COMMAND REFERENCE GUIDE

# Ruckus FastIron Command Reference Guide, 08.0.80

Supporting FastIron Software Release 08.0.80

*Part Number: 53-1005540-08*
*Publication Date: 03 September 2019*

# Copyright, Trademark and Proprietary Rights Information



## Export Restrictions

These products and associated technical data (in print or electronic form) may be subject to export control laws of the United States of America. It is your responsibility to determine the applicable regulations and to comply with them. The following notice is applicable for all products or technology subject to export control:

*These items are controlled by the U.S. Government and authorized for export only to the country of ultimate destination for use by the ultimate consignee or end-user(s) herein identified. They may not be resold, transferred, or otherwise disposed of, to any other country or to any person other than the authorized ultimate consignee or end-user(s), either in their original form or after being incorporated into other items, without first obtaining approval from the U.S. government or as otherwise authorized by U.S. law and regulations.*

## Disclaimer

THIS CONTENT AND ASSOCIATED PRODUCTS OR SERVICES ("MATERIALS"), ARE PROVIDED "AS IS" AND WITHOUT WARRANTIES OF ANY KIND, WHETHER EXPRESS OR IMPLIED. TO THE FULLEST EXTENT PERMISSIBLE PURSUANT TO APPLICABLE LAW, COMMSCOPE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, TITLE, NON-INFRINGEMENT, FREEDOM FROM COMPUTER VIRUS, AND WARRANTIES ARISING FROM COURSE OF DEALING OR COURSE OF PERFORMANCE. CommScope does not represent or warrant that the functions described or contained in the Materials will be uninterrupted or error-free, that defects will be corrected, or are free of viruses or other harmful components. CommScope does not make any warranties or representations regarding the use of the Materials in terms of their completeness, correctness, accuracy, adequacy, usefulness, timeliness, reliability or otherwise. As a condition of your use of the Materials, you warrant to CommScope that you will not make use thereof for any purpose that is unlawful or prohibited by their associated terms of use.

## Limitation of Liability

IN NO EVENT SHALL COMMSCOPE, COMMSCOPE AFFILIATES, OR THEIR OFFICERS, DIRECTORS, EMPLOYEES, AGENTS, SUPPLIERS, LICENSORS AND THIRD PARTY PARTNERS, BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER, EVEN IF COMMSCOPE HAS BEEN PREVIOUSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, WHETHER IN AN ACTION UNDER CONTRACT, TORT, OR ANY OTHER THEORY ARISING FROM YOUR ACCESS TO, OR USE OF, THE MATERIALS. Because some jurisdictions do not allow limitations on how long an implied warranty lasts, or the exclusion or limitation of liability for consequential or incidental damages, some of the above limitations may not apply to you.

## Trademarks

ARRIS, the ARRIS logo, CommScope, Ruckus, Ruckus Wireless, Ruckus Networks, Ruckus logo, the Big Dog design, BeamFlex, ChannelFly, EdgeIron, FastIron, HyperEdge, ICX, IronPoint, OPENG, SmartCell, Unleashed, Xclaim, and ZoneFlex are trademarks of CommScope, Inc. and/or its affiliates. Wi-Fi Alliance, Wi-Fi, the Wi-Fi logo, Wi-Fi Certified, the Wi-Fi CERTIFIED logo, Wi-Fi Protected Access, the Wi-Fi Protected Setup logo, Wi-Fi Protected Setup, Wi-Fi Multimedia and WPA2 and WMM are trademarks or registered trademarks of Wi-Fi Alliance. All other trademarks are the property of their respective owners.

# Contents

**Preface** ..... **41**
Document Conventions ..... 41
Notes, Cautions, and Warnings ..... 41
Command Syntax Conventions ..... 42
Document Feedback ..... 42
Ruckus Product Documentation Resources ..... 42
Online Training Resources ..... 43
Contacting Ruckus Customer Services and Support ..... 43
What Support Do I Need? ..... 43
Open a Case ..... 43
Self-Service Resources ..... 43

**About This Document** ..... **45**
What's new in this document ..... 45
New and modified commands for FastIron 08.0.80c ..... 45
New commands for FastIron 08.0.80 ..... 45
Modified commands for FastIron 08.0.80 ..... 46
Deprecated commands for FastIron 08.0.80 ..... 47
Other enhancements for FastIron 08.0.80 ..... 47
Supported hardware ..... 48

**Using the FastIron Command-Line Interface** ..... **49**
Accessing the CLI ..... 49
Command configuration modes ..... 49
Command help ..... 50
Command completion ..... 50
Scroll control ..... 51
Line editing commands ..... 52
Searching and filtering command output ..... 52
Searching and filtering output at the --More-- prompt ..... 52
Searching and filtering show command output ..... 53
Creating an alias for a CLI command ..... 56
Configuration notes for creating a command alias ..... 57

**Commands A and B** ..... **59**
100-fx ..... 59
100-tx ..... 60
aaa accounting commands ..... 61
aaa accounting dot1x ..... 63
aaa accounting exec ..... 65
aaa accounting mac-auth ..... 67
aaa accounting system ..... 69
aaa authentication dot1x ..... 71
aaa authentication enable ..... 73
aaa authentication login ..... 75
aaa authentication snmp-server ..... 77
aaa authentication web-server ..... 79
aaa authorization coa enable ..... 81

aaa authorization coa ignore ..... 82
aaa authorization commands..... 84
aaa authorization exec..... 86
accept-lifetime ..... 88
accept-mode..... 90
access-control vlan..... 92
accounting..... 93
acl-logging..... 94
acl-mirror-port..... 96
activate (VRRP)..... 98
activate (VSRP)..... 99
auto-enroll (PKI)..... 100
add mac..... 101
add-vlan..... 103
address-family..... 104
address-family unicast (BGP)..... 105
advertise backup..... 107
advertise backup (VSRP)..... 108
age..... 109
aggregate-address (BGP)..... 111
aggregated-vlan..... 113
alias..... 114
all-client..... 116
always-compare-med ..... 117
always-propagate ..... 118
anycast-rp..... 120
area authentication (IPsec) ..... 122
area authentication (OSPFv3) ..... 124
area authentication keychain (OSPFv3) ..... 126
area nssa (OSPFv2)..... 127
area nssa (OSPFv3)..... 129
area range (OSPFv2)..... 131
area range (OSPFv3)..... 133
area stub (OSPFv2)..... 135
area stub (OSPFv3)..... 136
area virtual-link (OSPFv2)..... 137
area virtual-link (OSPFv3)..... 139
area virtual-link authentication (OSPFv2)..... 141
area virtual-link authentication (OSPFv3)..... 143
area virtual-link authentication ipsec (OSPFv3)..... 145
area virtual-link authentication key-activation-wait-time (OSPFv2)..... 147
area virtual-link authentication key-activation-wait-time (OSPFv3)..... 149
area virtual-link authentication keychain (OSPFv2)..... 151
area virtual-link authentication keychain (OSPFv3)..... 153
area virtual-link authentication plain-text (OSPFv2)..... 155
area virtual-link authentication rfc6506 (OSPFv3)..... 157
arp..... 158
arp-internal-priority..... 160
as-path-ignore ..... 161
atalk-proto..... 162

attempt-max-num....163
auth allow-tagged enable....164
auth auth-mode....165
auth-default-vlan....166
auth-fail-action (flexible authentication)....168
auth-mode....170
auth-mode captive-portal....171
auth-mode none....172
auth-mode passcode....173
auth-mode username-password....176
auth-order....178
auth-timeout-action....180
auth-vlan-mode....182
authenticate....184
authenticated-mac-age-time....185
authentication....186
authentication (IKEv2)....187
authentication auth-default-vlan....188
authentication auth-order....190
authentication auth-vlan-mode....192
authentication disable-aging....194
authentication dos-protection....195
authentication fail-action....197
authentication filter-strict-security....199
authentication max-sessions....201
authentication reauth-timeout....203
authentication source-guard-protection enable....204
authentication timeout-action....206
authentication voice-vlan....208
authentication-algorithm....210
authentication-key....212
auto-cost reference-bandwidth (OSPFv2)....213
auto-cost reference-bandwidth (OSPFv3)....215
auto-lacp....217
autosave....218
backup....219
backup (VSRP)....221
backup-hello-interval....223
backup-hello-interval (VSRP)....224
bandwidth (interface)....225
banner....227
batch buffer....229
bgp-redistribute-internal....231
block....232
block-applicant....233
block-learning....235
boot system flash....237
boot system tftp....239
bootfile....240
bootp-relay-max-hops....241

bpdu-flood-enable ... 242
breakout ethernet ... 243
broadcast client ... 246
broadcast destination ... 247
broadcast limit (enable) ... 248
broadcast limit (logging) ... 249
bsr-candidate ... 250
bsr-msg-interval ... 252
buffer-profile port-region ... 253
buffer-sharing-full ... 255

**Commands C ... 257**
capability as4 ... 257
captive-portal ... 258
captive-portal profile ... 259
cdp enable ... 260
cdp run ... 261
chassis fanless ... 262
chassis name ... 264
clear access-list ... 265
clear access-list accounting ... 266
clear acl-on-arp ... 268
clear authentication sessions ... 269
clear authentication statistics ... 271
clear cable diagnostics tdr ... 272
clear cli-command-history ... 273
clear dhcp ... 274
clear dot1x sessions ... 275
clear dot1x statistics ... 276
clear dot1x-mka statistics ... 277
clear fdp counters ... 278
clear fdp table ... 279
clear gvrp statistics ... 280
clear ikev2 sa ... 281
clear ikev2 statistics ... 283
clear ip bgp dampening ... 284
clear ip bgp flap-statistics ... 285
clear ip bgp local routes ... 286
clear ip bgp neighbor ... 287
clear ip bgp routes ... 289
clear ip bgp traffic ... 290
clear ip bgp vrf ... 291
clear ip dhcp-server binding ... 292
clear ip dhcp-server statistics ... 293
clear ip igmp cache ... 294
clear ip igmp traffic ... 295
clear ip mroute ... 296
clear ip msdp peer ... 298
clear ip msdp sa-cache ... 299
clear ip msdp statistics ... 300
clear ip multicast counters ... 301

clear ip multicast mcache....302
clear ip multicast traffic....303
clear ip ospf....304
clear ip pim cache....306
clear ip pim counters....307
clear ip pim hw-resource....308
clear ip pim rp-map....309
clear ip pim traffic....310
clear ip pimsm-snoop....311
clear ip route....312
clear ip tunnel....313
clear ip vrrp statistics....314
clear ip vrrp-extended statistics....315
clear ipsec sa....316
clear ipv6 bgp dampening....317
clear ipv6 bgp flap-statistics....318
clear ipv6 bgp local routes....319
clear ipv6 bgp neighbor....320
clear ipv6 bgp routes....322
clear ipv6 bgp traffic....323
clear ipv6 cache....324
clear ipv6 dhcp6 snooping....326
clear ipv6 dhcp-relay delegated-prefixes....327
clear ipv6 dhcp-relay statistics....328
clear ipv6 mld traffic....329
clear ipv6 mroute....330
clear ipv6 multicast counters....331
clear ipv6 multicast mcache....332
clear ipv6 multicast traffic....333
clear ipv6 neighbor....334
clear ipv6 neighbor inspection....336
clear ipv6 ospf....338
clear ipv6 pim cache....340
clear ipv6 pim counters....341
clear ipv6 pim hw-resource....342
clear ipv6 pim rp-map....343
clear ipv6 pim traffic....344
clear ipv6 pimsm-snoop....345
clear ipv6 raguard....346
clear ipv6 rip route....347
clear ipv6 route....348
clear ipv6 traffic....349
clear ipv6 tunnel....350
clear ipv6 tunnel stat....351
clear ipv6 vrrp statistics....352
clear ipv6 vrrp-extended statistics....353
clear link-keepalive statistics....354
clear link-oam statistics....355
clear lldp neighbors....356
clear lldp statistics....357

clear logging....358
clear loop-detection....359
clear l2protocol dot1q-tunnel counters....360
clear mac-address....361
clear mac-address cluster....362
clear mac-authentication sessions....363
clear mac-authentication statistics....364
clear macsec statistics....365
clear management-vrf-stats....366
clear notification-mac statistics....367
clear openflow....368
clear port security....369
clear public-key....370
clear pvstplus-protect-statistics....371
clear stack ipc....372
clear statistics....374
clear statistics openflow....376
clear stp-protect-statistics....377
clear webauth vlan....378
clear web-connection....379
client....380
client-auto-detect config....381
client-auto-detect ethernet....382
client-auto-detect start....383
client-auto-detect stop....384
client-interface....385
client-interfaces shutdown....386
client-isolation....387
client-to-client-reflection....388
clock set....389
clock summer-time....390
clock timezone....392
cluster....394
cluster-id....395
compare-routerid....396
confederation identifier....397
confederation peers....398
console timeout....399
copy disk0....401
copy disk0 flash....402
copy flash disk0....404
copy flash flash....406
copy flash scp....408
copy flash tftp....410
copy https flash....411
copy https startup-config....413
copy running-config disk0....415
copy running-config https....416
copy running-config scp....417
copy running-config tftp....419

copy scp flash....420
copy scp license....423
copy scp running-config....425
copy scp startup-config....427
copy startup-config disk0....429
copy startup-config https....430
copy startup-config scp....431
copy startup-config tftp....433
copy tftp flash....434
copy tftp license....436
copy tftp running-config....437
copy tftp startup-config....438
copy tftp system-manifest....439
cpu-limit....442
critical-vlan....443
crl-query (PKI)....444
crl-update-time (PKI)....445
crypto key client generate....446
crypto key client zeroize....447
crypto key generate....448
crypto key zeroize....450
crypto-ssl certificate....451
cycle-time....452

**Commands D through H....453**

dampening....453
database-overflow-interval (OSPFv2)....455
database-overflow-interval (OSPFv3)....456
dead-interval....457
dead-interval (VSRP)....459
decnet-proto....460
default-acl....461
default-gateway....463
default-information-originate (BGP)....464
default-information-originate (OSPFv2)....465
default-information-originate (OSPFv3)....467
default-ipv6-gateway....469
default-local-preference....471
default-metric (BGP)....472
default-metric (OSPF)....473
default-metric (RIP)....474
default-passive-interface....475
default-ports....476
default-timers....478
default-vlan-id....479
delay-notifications....480
delete-all....481
deny (extended IPv4 ACLs)....482
deny (standard IPv4 ACLs)....487
description (IKEv2)....489
description (IPsec)....490

destination-ip....491
dhcp-default-router....492
dhcp-gateway-list....493
dhcp snooping client-learning disable....494
dhcp snooping relay information....495
dhcp snooping relay information circuit-id....496
dhcp snooping relay information remote-id....497
dhcp snooping relay information subscriber-id....498
dhcp snooping trust....499
dhcp6 snooping trust....500
dhgroup....501
diagnostics (MRP)....502
disable (LAG)....503
disable (NTP)....505
disable (Port)....506
disable (VSRP)....507
disable authentication md5....508
disable-aging....509
distance (BGP)....510
distance (OSPF)....511
distance (RIP)....513
distribute-list prefix-list (OSPFv3)....514
distribute-list prefix-list (RIPng)....516
distribute-list route-map....517
dlb-internal-trunk-hash....518
dns-filter....520
domain-name....522
dot1x auth-filter....523
dot1x enable....525
dot1x guest-vlan....527
dot1x initialize....528
dot1x macauth-override....529
dot1x max-reauth-req....530
dot1x max-req....531
dot1x-mka-enable....532
dot1x port-control....533
dot1x timeout....535
dynamic....537
eckeypair (PKI)....539
eee....540
egress-buffer-profile....542
enable (GVRP)....544
enable (LAG)....546
enable (MAC Port Security)....548
enable (MRP)....549
enable (Port)....550
enable (VSRP)....551
enable (Web Authentication)....552
enable aaa console....553
enable acl-per-port-per-vlan....555

enable egress-acl-on-cpu-traffic ..... 556
enable nd hop-limit ..... 557
enable password-display ..... 558
enable password-min-length ..... 559
enable port-config-password ..... 560
enable read-only-password ..... 561
enable snmp ..... 562
enable strict-password-enforcement ..... 563
enable super-user-password ..... 564
enable telnet ..... 565
enable user ..... 566
enable-accounting ..... 568
enable-mka ..... 569
encapsulation-mode ..... 571
encryption ..... 572
encryption-algorithm ..... 573
enforce-first-as ..... 575
enrollment (PKI) ..... 576
erase system factory-default ..... 577
erase flash ..... 579
erase startup-config ..... 580
errdisable packet-inerror-detect ..... 581
errdisable recovery ..... 582
esn-enable (IPsec) ..... 585
ethernet (EFM-OAM) ..... 586
ethernet loopback ..... 588
ethernet loopback (VLAN-aware) ..... 590
ethernet loopback test-mac ..... 592
exclude ethernet ..... 594
excluded-address ..... 595
execute batch ..... 596
extend vlan add (VXLAN) ..... 600
external-lsdb-limit (OSPFv2) ..... 601
external-lsdb-limit (OSPFv3) ..... 602
failover ..... 603
fast-external-fallover ..... 604
fast port-span ..... 605
fast uplink-span ..... 607
fdp advertise ..... 609
fdp enable ..... 610
fdp holdtime ..... 611
fdp run ..... 612
fdp timer ..... 613
fingerprint (PKI) ..... 614
filter-strict-security enable ..... 615
flash ..... 617
flash-timeout ..... 619
flow-control ..... 620
force-up ethernet ..... 622
format disk0 ..... 624

gig-default ... 625
graceful-restart (BGP) ... 626
graceful-restart (OSPFv2) ... 629
graceful-restart helper (OSPFv3) ... 631
graft-retransmit-timer ... 632
group-router-interface ... 633
gvrp-base-vlan-id ... 634
gvrp-enable ... 635
gvrp-max-leaveall-timer ... 636
hardware-drop-disable ... 637
hello-interval (VRRP) ... 638
hello-interval (VSRP) ... 640
hello-timer ... 642
hitless-failover enable ... 643
hold-down-interval ... 644
host-max-num ... 645
hostname ... 646

**Commands I ... 647**

ignore-temp-shutdown ... 647
ike-profile ... 649
ikev2 auth-proposal ... 650
ikev2 exchange-max-time ... 651
ikev2 limit ... 652
ikev2 policy ... 653
ikev2 profile ... 655
ikev2 proposal ... 657
ikev2 retransmit-interval ... 659
ikev2 retry-count ... 660
image-auto-copy disable ... 661
import-users ... 662
inactivity-timer ... 663
include-port ... 664
initial-contact-payload ... 665
initial-ttl ... 666
inline power ... 667
inline power adjust class ... 669
inline power couple-datalink ... 671
inline power install-firmware ... 673
inline power install-firmware scp ... 675
inline power interface-mode-2pair-pse ... 677
inline power non-pd-detection enable ... 679
inline power overdrive ... 681
integrity ... 682
interface ethernet ... 683
interface group-ve ... 684
interface lag ... 685
interface loopback ... 686
interface management ... 687
interface tunnel ... 688
interface ve ... 689

ip access-group....690
ip access-list....692
ip address....695
ip-address....697
ip-address (VSRP)....699
ip arp inspection syslog disable....700
ip arp inspection validate....701
ip arp inspection vlan....703
ip arp learn-gratuitous-arp....705
ip arp port-move-syslog....706
ip arp-age....707
ip bootp-gateway....708
ip bootp-use-intf-ip....709
ip broadcast-zero....710
ip default-gateway....711
ip default-network....712
ip dhcp-client auto-update enable....713
ip dhcp-client enable....714
ip dhcp-server arp-ping-timeout....715
ip dhcp-server enable....716
ip dhcp-server mgmt....717
ip dhcp-server pool....718
ip dhcp-server relay-agent-echo enable....719
ip dhcp-server server-identifier....720
ip dhcp snooping relay information disable....721
ip dhcp snooping vlan....723
ip dhcp relay information policy....725
ip directed-broadcast....726
ip dns....727
ip dscp-remark....728
ip encapsulation....729
ip follow ve....730
ip forward-protocol udp....731
ip helper-address....732
ip helper-use-responder-ip....734
ip hitless-route-purge-timer....735
ip icmp burst-normal....736
ip icmp echo broadcast-request....738
ip icmp redirects....739
ip icmp unreachable....740
ip igmp group-membership-time....742
ip igmp max-group-address....743
ip igmp max-response-time....744
ip igmp port-version....745
ip igmp proxy....746
ip igmp query-interval....748
ip igmp ssm-map....749
ip igmp static-group....750
ip igmp tracking....752
ip igmp version....753

ip interface loopback (VXLAN)....754
ip irdp....755
ip irdp (interface)....756
ip load-sharing....758
ip-mac....759
ip max-mroute....760
ip mroute....761
ip mroute (next hop)....763
ip mroute next-hop-enable-default....765
ip mroute next-hop-recursion....766
ip mtu....768
ip multicast....769
ip multicast age-interval....771
ip multicast disable-flooding....772
ip multicast leave-wait-time....773
ip multicast max-response-time....774
ip multicast mcache-age....775
ip multicast optimization....776
ip multicast query-interval....777
ip multicast report-control....778
ip multicast verbose-off....779
ip multicast version....780
ip multicast-boundary....781
ip multicast-debug-mode....782
ip multicast-nonstop-routing....783
ip multicast-routing optimization....784
ip multicast-routing rpf-check mac-movement....785
ip ospf active....786
ip ospf area....787
ip ospf authentication....788
ip ospf authentication key-activation-wait-time....790
ip ospf authentication keychain....791
ip ospf authentication plain-text....792
ip ospf cost....793
ip ospf database-filter....794
ip ospf dead-interval....796
ip ospf hello-interval....797
ip ospf mtu-ignore....798
ip ospf network....799
ip ospf passive....801
ip ospf priority....802
ip ospf retransmit-interval....803
ip ospf transmit-delay....804
ip pcp-remark....805
ip pim....806
ip pim border....808
ip pim dr-priority....809
ip pim neighbor-filter....810
ip pim-sparse....812
ip pimsm-snooping....813

ip policy route-map ..... 814
ip prefix-list ..... 815
ip preserve-acl-user-input-format ..... 817
ip-proto ..... 819
ip proxy-arp ..... 820
ip proxy-arp (interface) ..... 821
ip radius source-interface ..... 822
ip rarp ..... 824
ip redirect ..... 825
ip rip ..... 826
ip rip metric-offset ..... 827
ip rip prefix-list ..... 829
ip rip route-map ..... 830
ip route ..... 831
ip route next-hop ..... 834
ip route next-hop-enable-default ..... 835
ip route next-hop-recursion ..... 836
ip router-id ..... 837
ip show-portname ..... 838
ip show-service-number-in-log ..... 839
ip show-subnet-length ..... 840
ip source-route ..... 841
ip ssh authentication-retries ..... 842
ip ssh client ..... 843
ip ssh encryption aes-only ..... 845
ip ssh encryption disable-aes-cbc ..... 846
ip ssh idle-time ..... 847
ip ssh interactive-authentication ..... 848
ip ssh key-authentication ..... 849
ip ssh key-exchange-method dh-group1-sha1 ..... 850
ip ssh password-authentication ..... 851
ip ssh permit-empty-password ..... 852
ip ssh port ..... 853
ip ssh pub-key-file ..... 854
ip ssh rekey ..... 856
ip ssh scp ..... 858
ip ssh strict-management-vrf ..... 859
ip ssh timeout ..... 861
ip ssl ..... 862
ip ssl min-version ..... 865
ip-subnet ..... 866
ip syslog source-interface ..... 867
ip tacacs source-interface ..... 869
ip tcp burst-normal ..... 871
ip tcp keepalive ..... 873
ip-telephony data ..... 874
ip-telephony voice ..... 876
ip telnet source-interface ..... 878
ip tftp source-interface ..... 880
ip ttl ..... 882

ip use-acl-on-arp....883
ip vrrp auth-type....885
ip vrrp vrid....887
ip vrrp-extended auth-type....888
ip vrrp-extended vrid....890
ipsec profile....891
ipsec proposal....892
ipv6 access-list....893
ipv6 address....894
ipv6 cache-lifetime....895
ipv6 default-gateway....896
ipv6 dhcp-relay destination....898
ipv6 dhcp-relay distance....900
ipv6 dhcp-relay include-options....901
ipv6 dhcp-relay maximum-delegated-prefixes....903
ipv6 dhcp6 snooping vlan....905
ipv6 dns server-address....907
ipv6 enable....908
ipv6 hitless-route-purge-timer....909
ipv6 hop-limit....910
ipv6 icmp error-interval....911
ipv6 icmp fragment_header_bit....912
ipv6 icmp source-route....913
ipv6 load-sharing....914
ipv6 max-mroute....915
ipv6 mld group-membership-time....916
ipv6 mld llqi....917
ipv6 mld max-group-address....918
ipv6 mld max-response-time....919
ipv6 mld port-version....920
ipv6 mld query-interval....921
ipv6 mld robustness....922
ipv6 mld static-group....923
ipv6 mld tracking....925
ipv6 mld version....926
ipv6 mroute....928
ipv6 mroute (next hop)....930
ipv6 mroute next-hop-enable-default....932
ipv6 mroute next-hop-recursion....934
ipv6 mtu....936
ipv6 multicast....937
ipv6 multicast age-interval....938
ipv6 multicast disable-flooding....939
ipv6 multicast leave-wait-time....940
ipv6 multicast max-response-time....941
ipv6 multicast mcache-age....942
ipv6 multicast optimization....944
ipv6 multicast query-interval....945
ipv6 multicast report-control....946
ipv6 multicast verbose-off....947

ipv6 multicast version ..... 948
ipv6 multicast-boundary ..... 949
ipv6 multicast-routing optimization ..... 950
ipv6 multicast-routing rpf-check mac-movement ..... 951
ipv6 nd dad attempts ..... 952
ipv6 nd managed-config-flag ..... 953
ipv6 nd ns-interval ..... 954
ipv6 nd other-config-flag ..... 955
ipv6 nd prefix-advertisement ..... 956
ipv6 nd ra-dns-server ..... 958
ipv6 nd ra-domain-name ..... 959
ipv6 nd ra-hop-limit ..... 960
ipv6 nd ra-interval ..... 961
ipv6 nd ra-lifetime ..... 963
ipv6 nd reachable-time ..... 964
ipv6 nd router-preference ..... 965
ipv6 nd suppress-ra ..... 966
ipv6 nd suppress-ra address ..... 967
ipv6 neighbor ..... 969
ipv6 neighbor inspection ..... 971
ipv6 neighbor inspection vlan ..... 973
ipv6 ospf active ..... 975
ipv6 ospf area ..... 976
ipv6 ospf authentication ..... 977
ipv6 ospf authentication disable ..... 979
ipv6 ospf authentication ipsec ..... 980
ipv6 ospf authentication ipsec disable ..... 981
ipv6 ospf authentication ipsec spi ..... 982
ipv6 ospf authentication key-activation-wait-time ..... 984
ipv6 ospf authentication keychain ..... 985
ipv6 ospf authentication rfc6506 ..... 986
ipv6 ospf cost ..... 987
ipv6 ospf dead-interval ..... 988
ipv6 ospf hello-interval ..... 989
ipv6 ospf hello-jitter ..... 990
ipv6 ospf instance ..... 991
ipv6 ospf mtu-ignore ..... 992
ipv6 ospf network ..... 993
ipv6 ospf passive ..... 994
ipv6 ospf priority ..... 995
ipv6 ospf retransmit-interval ..... 996
ipv6 ospf suppress-linklsa ..... 997
ipv6 ospf transmit-delay ..... 998
ipv6 pim border ..... 999
ipv6 pim dr-priority ..... 1000
ipv6 pim neighbor-filter ..... 1001
ipv6 pim-sparse ..... 1003
ipv6 pimsm-snooping ..... 1004
ipv6 policy route-map ..... 1005
ipv6 prefix-list ..... 1007

ipv6-proto....1009
ipv6 raguard policy....1010
ipv6 raguard vlan....1011
ipv6 raguard whitelist....1012
ipv6 redirects....1013
ipv6 rip default-information....1014
ipv6 rip enable....1015
ipv6 rip metric-offset....1016
ipv6 rip summary-address....1017
ipv6 route....1018
ipv6 route next-hop....1020
ipv6 route next-hop-enable-default....1021
ipv6 route next-hop-recursion....1022
ipv6 router ospf....1023
ipv6 router pim....1024
ipv6 router rip....1025
ipv6 router vrrp....1026
ipv6 router vrrp-extended....1027
ipv6 traffic-filter....1028
ipv6 unicast-routing....1029
ipv6 vrrp vrid....1030
ipv6 vrrp-extended vrid....1031
ipv6-address....1032
ipv6-address auto-gen-link-local....1034
ipv6-neighbor inspection trust....1035
ipx-network....1036
ipx-proto....1038
issu abort....1039
issu primary....1040
issu secondary....1042

**Commands J, K, and L....1045**

jitc enable....1045
jitc show....1046
join-timer leave-timer leaveall-timer....1047
jumbo....1049
keep-alive-vlan....1050
keepalive....1051
keepalive (IKEv2)....1052
keychain....1053
key-id....1054
key-rollover-interval....1056
key-server-priority....1057
kill....1059
lacp-timeout....1060
lag....1061
lacp-mode passive....1063
lag-mac....1064
learn-default....1065
lease....1067
legacy-inline-power....1068

legacy-inline-power (interface)..... 1070
license delete perpetual..... 1071
license delete unit..... 1073
license install perpetual..... 1075
license set serial-number..... 1077
lifetime (IKEv2)..... 1079
lifetime (IPsec)..... 1080
link-config gig copper autoneg-control..... 1081
link-error-disable..... 1083
link-fault-signal..... 1085
link-keepalive ethernet..... 1086
link-keepalive interval..... 1088
link-keepalive retries..... 1089
link-oam..... 1090
lldp advertise link-aggregation..... 1091
lldp advertise mac-phy-config-status..... 1093
lldp advertise management-address..... 1095
lldp advertise max-frame-size..... 1097
lldp advertise med-capabilities..... 1099
lldp advertise med-power-via-mdi..... 1101
lldp advertise port-description..... 1103
lldp advertise port-id-subtype..... 1104
lldp advertise port-vlan-id..... 1106
lldp advertise power-via-mdi..... 1107
lldp advertise system-capabilities..... 1109
lldp advertise system-description..... 1110
lldp advertise system-name..... 1111
lldp enable ports..... 1112
lldp enable receive..... 1113
lldp enable snmp med-topo-change-notifications..... 1115
lldp enable snmp notifications..... 1116
lldp enable transmit..... 1117
lldp max-neighbors-per-port..... 1119
lldp max-total-neighbors..... 1120
lldp med fast-start-repeat-count..... 1121
lldp med location-id civic-address..... 1122
lldp med location-id coordinate-based..... 1126
lldp med location-id ecs-elin..... 1128
lldp med network-policy application..... 1129
lldp reinit-delay..... 1132
lldp run..... 1133
lldp snmp-notification-interval..... 1134
lldp tagged-packets..... 1135
lldp transmit-delay..... 1136
lldp transmit-hold..... 1137
lldp transmit-interval..... 1138
load-balance symmetric..... 1139
local-as ..... 1140
local-certificate (PKI)..... 1141
local-identifier..... 1142

local-userdb ... 1144
log (OSPFv2) ... 1145
logging ... 1147
logging buffered ... 1148
logging console ... 1150
logging cli-command ... 1151
logging-enable ... 1152
logging enable config-changed ... 1153
logging enable ikev2 ... 1154
logging enable ipsec ... 1155
logging enable (PKI) ... 1156
logging enable rfc5424 ... 1157
logging enable user-login ... 1158
logging facility ... 1159
logging host ... 1161
logging on ... 1162
logging persistence ... 1163
login-page ... 1164
log-status-change ... 1166
loop-detection ... 1167
loop-detection-interval ... 1168
loop-detection shutdown-disable ... 1169
loop-detection-syslog-interval ... 1170
l2protocol dot1q-tunnel ... 1171
l2protocol dot1q-tunnel cos ... 1173
l2protocol dot1q-tunnel drop-threshold ... 1174
l2protocol dot1q-tunnel-mac ... 1176
l2protocol dot1q-tunnel shutdown-threshold ... 1177

**Commands M ... 1179**

mac-age-time ... 1179
mac-authentication auth-filter ... 1181
mac-authentication dot1x-disable ... 1182
mac-authentication dot1x-override ... 1183
mac-authentication enable (Flexible authentication) ... 1184
mac-authentication password-format ... 1186
mac-authentication password-override (Flexible authentication) ... 1188
mac filter ... 1189
mac filter enable-accounting ... 1191
mac filter log-enable ... 1192
mac filter-group ... 1193
mac filter-group log-enable ... 1194
mac-learn-disable ... 1195
mac-notification interval ... 1196
mac-movement notification ... 1197
macsec cipher-suite ... 1198
macsec confidentiality-offset ... 1200
macsec frame-validation ... 1202
macsec replay-protection ... 1204
management exclude ... 1206
management-vlan ... 1208

management-vrf....1209
map vlan (vxlan)....1210
master....1212
master (MRP)....1213
master-vlan....1214
master-vlan (STP)....1215
match address-local....1216
match as-path....1218
match community....1219
match fvrf....1220
match-identity....1221
match interface....1223
match ip address....1224
match ipv6 address....1226
match metric....1227
match protocol....1228
match route-type....1229
match tag....1230
maxas-limit....1231
maximum (port MAC security)....1232
maximum-paths (BGP)....1233
maximum-paths ebgp ibgp....1235
maximum-preference....1237
max-hw-age....1238
max-mcache....1239
max-metric router-lsa (OSPFv2)....1240
max-sessions....1242
max-sw-age....1244
max-vlan (SPX)....1245
max-vlans-per-pe-port (SPX)....1247
med-missing-as-worst....1249
member-group....1250
member-group (STP)....1251
member-vlan....1252
member-vlan (STP)....1253
mesh-group....1254
message-interval....1255
metric-type....1256
metro-ring....1257
mdi-mdix....1258
mirror-port....1260
mka-cfg-group....1261
module (SPX)....1263
monitor (ERSPAN)....1264
monitor (LAG)....1266
monitor....1268
monitor-profile....1270
mount disk0....1272
msdp-peer....1273
mstp admin-edge-port....1275

mstp admin-pt2pt-mac....1276
mstp disable....1277
mstp edge-port-auto-detect....1278
mstp force-migration-check....1279
mstp force-version....1280
mstp forward-delay....1281
mstp hello-time....1282
mstp instance....1283
mstp max-age....1285
mstp max-hops....1286
mstp name....1287
mstp revision....1288
mstp root-protect timeout....1289
mstp scope....1290
mstp start....1291
mtu-exceed....1292
multicast disable-igmp-snoop....1293
multicast disable-pimsm-snoop....1294
multicast fast-convergence....1295
multicast fast-leave-v2....1296
multicast limit (enable)....1297
multicast limit (logging)....1298
multicast pimsm-snooping....1299
multicast port-version....1301
multicast proxy-off....1303
multicast querier-address....1304
multicast6 querier-address....1305
multicast router-port....1306
multicast static-group....1308
multicast tracking....1310
multicast version....1311
multicast6....1312
multicast6 disable-mld-snoop....1313
multicast6 disable-pimsm-snoop....1314
multicast6 fast-convergence....1315
multicast6 fast-leave-v1....1316
multicast6 pimsm-snooping....1317
multicast6 port-version....1318
multicast6 proxy-off....1320
multicast6 router-port....1321
multicast6 static-group....1323
multicast6 tracking....1325
multicast6 version....1326
multipath....1327
multi-spx-lag....1329
multi-spx-port....1332
multi-stack-port....1335
multi-stack-trunk....1336

**Commands N....1337**

name (MRP)....1337

nbr-timeout....1338
neighbor (RIP)....1339
neighbor activate....1341
neighbor advertisement-interval....1343
neighbor allowas-in....1345
neighbor as-override....1347
neighbor capability as4....1348
neighbor capability orf prefixlist....1349
neighbor default-originate....1351
neighbor description....1352
neighbor ebgp-btsh....1354
neighbor ebgp-multihop....1356
neighbor enforce-first-as....1357
neighbor filter-list....1358
neighbor local-as....1360
neighbor maxas-limit in....1361
neighbor maximum-prefix....1362
neighbor next-hop-self....1363
neighbor password....1365
neighbor peer-group....1367
neighbor prefix-list....1369
neighbor remote-as....1371
neighbor remove-private-as....1373
neighbor route-map....1375
neighbor route-reflector-client....1376
neighbor send-community....1377
neighbor shutdown....1378
neighbor soft-reconfiguration inbound....1380
neighbor timers....1382
neighbor update-source....1384
neighbor weight....1386
netbios-name-server....1387
netbios-proto....1388
network....1389
network (dhcp)....1391
next-bootstrap-server....1392
next-hop-enable-default....1393
next-hop-recursion....1394
no-dynamic-aging....1395
non-preempt-mode....1396
non-preempt-mode (VRRP)....1397
nonstop-routing (OSPF)....1398
ntp....1399
ntp-interface....1400

**Commands O, P, Q, R, and Sa through Si....1403**

ocsp (PKI)....1403
ocsp-url (PKI)....1405
openflow enable....1406
openflow purge-time....1407
optical-monitor....1408

option ..... 1409
originator-id ..... 1411
other-proto ..... 1412
overlay-gateway ..... 1413
owner ..... 1414
password ..... 1416
packet-inerror-detect ..... 1417
pass-through ..... 1418
pdu-rate (EFM-OAM) ..... 1419
pe-id ..... 1420
pe-name ..... 1422
peer ..... 1423
peer disable-fast-failover ..... 1425
peer timers ..... 1426
peer-info ..... 1427
permit (extended IPv4 ACLs) ..... 1428
permit (standard IPv4 ACLs) ..... 1433
phy cable diagnostics tdr ..... 1435
phy-fifo-depth ..... 1436
ping ..... 1437
pki authenticate ..... 1439
pki cert-validate ..... 1440
pki enroll ..... 1441
pki-entity ..... 1442
pki profile-enrollment ..... 1443
pki trustpoint ..... 1444
poison-local-routes ..... 1446
poison-reverse ..... 1447
port security ..... 1449
port-down-authenticated-mac-cleanup ..... 1450
port-name ..... 1451
port-name (LAG) ..... 1452
port-statistics-reset-timestamp enable ..... 1453
ports ..... 1454
pre-shared-key ..... 1456
prefix-list ..... 1458
prefix-list (RIP) ..... 1459
preforwarding-time ..... 1460
prf ..... 1462
priority ..... 1463
priority-flow-control ..... 1464
priority-flow-control enable ..... 1466
privilege ..... 1468
profile-config ..... 1470
proposal (IKEv2) ..... 1472
proposal (ipsec) ..... 1473
protected ..... 1474
protected-port ..... 1475
prune-timer ..... 1477
prune-wait ..... 1478

pvlan mapping....1479
pvlan pvlan-trunk....1481
pvlan type....1483
pvst-mode....1485
pvstplus-protect....1486
qd-buffer....1487
qd-descriptor....1488
qos egress-buffer-profile port-share-level....1489
qos egress-buffer-profile queue-share-level....1491
qos egress-shape-ifg-bytes....1494
qos ingress-buffer-profile....1495
qos mechanism....1497
qos monitor-queue-drop-counters....1498
qos name....1499
qos priority-to-pg....1500
qos profile....1503
qos scheduler-profile....1505
qos sflow-set-cpu-rate-limit....1508
qos tagged-priority....1510
qos-internal-trunk-queue....1511
qos-tos map dscp-priority....1513
radius-client coa host....1515
radius-client coa port....1516
radius-server accounting....1517
radius-server dead-time....1518
radius-server enable....1519
radius-server host....1520
radius-server key....1524
radius-server retransmit....1525
radius-server test....1526
radius-server timeout....1527
raguard....1528
rarp....1529
rate-limit-arp....1530
rate-limit input....1531
rate-limit output....1532
rate-limit-log....1533
rconsole....1534
rconsole (SPX)....1535
rd....1537
rear-module....1538
re-authentication (Flexible authentication)....1540
reauth-period....1541
reauth-time....1542
redistribute....1543
redistribute (BGP)....1546
redistribute (RIP)....1548
redistribute (RIPng)....1550
regenerate-seq-num....1551
register-probe-time....1553

register-suppress-time ..... 1554
relative-utilization ..... 1555
reload ..... 1557
remark ..... 1559
remote-identifier ..... 1561
remote-loopback ..... 1563
remove-tagged-ports ..... 1565
remove-untagged-ports ..... 1567
remove-vlan(VLAN group) ..... 1569
replay-protection ..... 1570
reserved-vlan-map ..... 1571
responder-only ..... 1572
restart-ports ..... 1573
restart-vsrp-port ..... 1574
restricted-vlan ..... 1575
reverse-manifest-enable ..... 1576
reverse-path-check ..... 1577
revocation-check (PKI) ..... 1579
rfc1583-compatibility (OSPF) ..... 1580
ring-interfaces ..... 1581
rmon alarm ..... 1583
rmon event ..... 1585
rmon history ..... 1587
route-only ..... 1589
route-precedence ..... 1591
route-precedence admin-distance ..... 1593
router bgp ..... 1594
router msdp ..... 1595
router ospf ..... 1596
router pim ..... 1597
router rip ..... 1599
router vrrp ..... 1600
router vrrp-extended ..... 1601
router vsrp ..... 1602
router-interface ..... 1603
rpf-mode ..... 1604
rp-address ..... 1606
rp-adv-interval ..... 1608
rp-candidate ..... 1609
rp-embedded ..... 1611
rsakeypair (PKI) ..... 1612
rspan destination ..... 1613
rspan source ..... 1614
rspan-vlan ..... 1616
sa-filter ..... 1617
save-current-values ..... 1619
scp ..... 1620
secure-login ..... 1621
secure-mac-address ..... 1622
server (NTP) ..... 1623

service local-user-protection ..... 1624
service password-encryption ..... 1626
set interface null0 ..... 1628
set next-hop-ip-tunnel ..... 1629
scale-timer ..... 1631
scale-timer ..... 1633
scheduler-profile ..... 1634
sequence (permit | deny in extended IPv4 ACLs) ..... 1636
sequence (permit | deny in IPv6 ACLs) ..... 1642
sequence (permit | deny in standard IPv4 ACLs) ..... 1648
set ip next-hop ..... 1650
send-lifetime ..... 1652
sflow agent-ip ..... 1654
sflow destination ..... 1655
sflow enable ..... 1657
sflow export ..... 1658
sflow forwarding ..... 1660
sflow forwarding (LAG) ..... 1661
sflow management-vrf-disable ..... 1662
sflow max-packet-size ..... 1663
sflow polling-interval ..... 1664
sflow sample ..... 1665
sflow source ..... 1667
sflow source-port ..... 1669
sflow version ..... 1670
short-path-forwarding ..... 1671
site (vxlan) ..... 1672

**Show Commands ..... 1673**

show 802-1w ..... 1673
show aaa ..... 1675
show access-list ..... 1677
show access-list accounting ..... 1679
show access-list named-acl ..... 1684
show acl-on-arp ..... 1685
show arp ..... 1687
show authentication acls ..... 1689
show authentication configuration ..... 1691
show authentication sessions ..... 1693
show authentication statistics ..... 1696
show boot-monitor ..... 1698
show batch schedule ..... 1699
show boot-preference ..... 1700
show breakout ..... 1701
show cable-diagnostics tdr ..... 1703
show captive-portal ..... 1705
show chassis ..... 1707
show cli-command-history ..... 1710
show clock ..... 1712
show configuration ..... 1713
show configuration (SPX) ..... 1716

show cpu histogram....1718
show cpu-utilization....1723
show errdisable....1725
show default....1727
show default values....1730
show dlb-internal-trunk-hash....1731
show dot1x....1732
show dot1x configuration....1734
show dot1x ip-acl....1737
show dot1x mac-address-filter....1739
show dot1x mac-filter....1740
show dot1x mac-session....1741
show dot1x sessions....1743
show dot1x sessions detail....1745
show dot1x statistics....1747
show dot1x-mka config....1749
show dot1x-mka config-group....1751
show dot1x-mka sessions....1753
show dot1x-mka statistics....1756
show eee-statistics....1758
show eee-statistics ethernet....1760
show erspan....1761
show ethernet loopback interfaces....1763
show ethernet loopback resources....1765
show fdp entry....1766
show fdp interface....1768
show fdp neighbors....1769
show fdp traffic....1771
show files....1772
show files disk0....1773
show flash....1774
show gvrp....1775
show gvrp ethernet....1777
show gvrp statistics....1779
show gvrp vlan....1781
show hardware ipv6-route....1783
show hardware mac-entry....1784
show hardware nexthop usage....1785
show hardware route....1786
show ikev2....1788
show ikev2 auth-proposal....1790
show ikev2 policy....1792
show ikev2 profile....1794
show ikev2 proposal....1796
show ikev2 sa....1798
show ikev2 session....1801
show ikev2 statistics....1804
show inline power....1806
show inline power debug-info....1810
show inline power detail....1811

show inline power emesg....1816
show interfaces ethernet....1818
show interfaces lag....1821
show interfaces management....1824
show interfaces stack-ports....1825
show interfaces tunnel....1827
show interfaces ve....1829
show ip....1830
show ip access-lists....1833
show ip arp inspection entries....1834
show ip bgp....1836
show ip bgp attribute-entries....1837
show ip bgp config....1839
show ip bgp dampened-paths....1840
show ip bgp filtered-routes....1841
show ip bgp flap-statistics....1842
show ip bgp ipv6....1844
show ip bgp neighbors....1847
show ip bgp neighbors advertised-routes....1853
show ip bgp neighbors flap-statistics....1854
show ip bgp neighbors last-packet-with-error....1855
show ip bgp neighbors received....1856
show ip bgp neighbors received-routes....1857
show ip bgp neighbors rib-out-routes....1858
show ip bgp neighbors routes....1859
show ip bgp neighbors routes-summary....1860
show ip bgp peer-group....1863
show ip bgp routes....1864
show ip bgp routes community....1868
show ip bgp summary....1869
show ip bgp vrf....1872
show ip bgp vrf neighbors....1874
show ip bgp vrf routes....1876
show ip bgp vrf routes community....1878
show ip cache....1879
show ip client-pub-key....1881
show ip dhcp-client options....1882
show ip dhcp relay information....1885
show ip dhcp relay information brief....1886
show ip dhcp-server address-pools....1887
show ip dhcp-server binding....1889
show ip dhcp-server flash....1890
show ip dhcp-server statistics....1891
show ip dhcp-server summary....1892
show ip dhcp snooping flash....1893
show ip dhcp snooping info....1894
show ip dhcp snooping vlan....1895
show ip igmp group....1896
show ip igmp interface....1898
show ip igmp proxy....1900

show ip igmp settings....1903
show ip igmp ssm-map....1905
show ip igmp static....1907
show ip igmp traffic....1908
show ip interface....1910
show ip mroute....1913
show ip msdp mesh-group....1915
show ip msdp peer....1917
show ip msdp rpf-peer....1918
show ip msdp sa-cache....1919
show ip msdp summary....1922
show ip multicast....1924
show ip multicast error....1925
show ip multicast group....1926
show ip multicast mcache....1929
**show ip multicast optimization** ....1931
show ip multicast pimsm-snooping....1932
show ip multicast resource....1933
show ip multicast vlan....1934
show ip ospf....1940
show ip ospf area ....1941
show ip ospf border-routers ....1943
show ip ospf config ....1944
show ip ospf database ....1947
show ip ospf interface ....1951
show ip ospf neighbor ....1955
show ip ospf redistribute route ....1958
show ip ospf routes ....1959
show ip ospf summary ....1961
show ip ospf traffic ....1962
show ip ospf trap ....1963
show ip ospf virtual link ....1964
show ip ospf virtual neighbor ....1965
show ip multicast traffic....1966
show ip pim bsr....1968
show ip pim counter nsr....1970
show ip pim dense....1972
show ip pim group....1974
show ip pim hw-resource....1976
show ip pim interface....1977
show ip pim mcache....1978
show ip pim neighbor....1984
show ip pim nsr....1986
show ip pim optimization....1987
show ip pim prune....1989
show ip pim resource....1990
show ip pim rp-candidate....1992
show ip pim rpf....1994
show ip pim rp-hash....1995
show ip pim rp-map....1996

show ip pim rp-set ... 1997
show ip pim sparse ... 1999
show ip pim traffic ... 2002
show ip pimsm-snooping cache ... 2005
show ip reverse-path-check ... 2008
show ip reverse-path-check interface ... 2009
show ip rip ... 2010
show ip rip interface ... 2012
show ip rip route ... 2014
show ip route ... 2015
show ip source-guard ... 2017
show ip ssh ... 2019
show ip ssl (FIPS) ... 2022
show ip static mroute ... 2026
show ip static-arp ... 2027
show ip traffic ... 2028
show ip tunnel traffic ... 2031
show ip vrrp ... 2032
show ip vrrp-extended ... 2035
show ipc_stats ... 2039
show ipsec card-utilization ... 2040
show ipsec profile ... 2042
show ipsec proposal ... 2044
show ipsec sa ... 2046
show ipv6 ... 2050
show ipv6 access-list ... 2051
show ipv6 bgp ... 2053
show ipv6 bgp attribute-entries ... 2055
show ipv6 bgp config ... 2057
show ipv6 bgp dampened-paths ... 2058
show ipv6 bgp filtered-routes ... 2059
show ipv6 bgp flap-statistics ... 2062
show ipv6 bgp neighbors ... 2064
show ipv6 bgp neighbors advertised-routes ... 2070
show ipv6 bgp neighbors flap-statistics ... 2072
show ipv6 bgp neighbors last-packet-with-error ... 2073
show ipv6 bgp neighbors received ... 2074
show ipv6 bgp neighbors received-routes ... 2075
show ipv6 bgp neighbors rib-out-routes ... 2078
show ipv6 bgp neighbors routes ... 2080
show ipv6 bgp neighbors routes-summary ... 2083
show ipv6 bgp peer-group ... 2085
show ipv6 bgp routes ... 2086
show ipv6 bgp routes community ... 2090
show ipv6 bgp summary ... 2091
show ipv6 cache ... 2094
show ipv6 dhcp-relay ... 2096
show ipv6 dhcp-relay delegated-prefixes ... 2097
show ipv6 dhcp-relay destinations ... 2098
show ipv6 dhcp-relay interface ... 2099

show ipv6 dhcp-relay options....2100
show ipv6 dhcp-relay prefix-delegation-information....2101
show ipv6 dhcp6 snooping vlan....2102
show ipv6 dhcp6 snooping info....2103
show ipv6 fragment-header....2105
show ipv6 mld group....2106
show ipv6 mld interface....2108
show ipv6 mld settings....2110
show ipv6 mld static....2112
show ipv6 mld traffic....2113
show ipv6 mroute....2115
show ipv6 multicast....2117
show ipv6 multicast error....2118
show ipv6 multicast group....2119
show ipv6 multicast mcache....2121
**show ipv6 multicast optimization** ....2122
show ipv6 multicast pimsm-snooping....2123
show ipv6 multicast resource....2124
show ipv6 multicast traffic....2126
show ipv6 multicast vlan....2128
show ipv6 neighbor....2130
show ipv6 neighbor inspection....2132
show ipv6 ospf....2135
show ipv6 ospf area ....2136
show ipv6 ospf database ....2138
show ipv6 ospf interface ....2144
show ipv6 ospf memory ....2149
show ipv6 ospf neighbor ....2151
show ipv6 ospf redistribute route ....2155
show ipv6 ospf routes ....2157
show ipv6 ospf spf ....2159
show ipv6 ospf summary ....2163
show ipv6 ospf virtual-links ....2164
show ipv6 ospf virtual-neighbor ....2166
show ipv6 pim anycast-rp....2168
show ipv6 pim bsr....2169
show ipv6 pim counter....2171
show ipv6 pim group....2172
show ipv6 pim hw-resource....2173
show ipv6 pim interface....2174
show ipv6 pim mcache....2175
show ipv6 pim resource....2178
show ipv6 pim rp-candidate....2180
show ipv6 pim rpf....2182
show ipv6 pim rp-hash....2183
show ipv6 pim rp-map....2184
show ipv6 pim rp-set....2185
show ipv6 pim sparse....2187
show ipv6 pim traffic....2190
show ipv6 pimsm-snooping cache....2192

show ipv6 raguard ..... 2194
show ipv6 rip ..... 2196
show ipv6 rip route ..... 2197
show ipv6 route ..... 2199
show ipv6 router ..... 2201
show ipv6 static mroute ..... 2203
show ipv6 tcp connections ..... 2204
show ipv6 tcp status ..... 2206
show ipv6 traffic ..... 2208
show ipv6 tunnel ..... 2212
show ipv6 tunnel traffic ..... 2213
show ipv6 vrrp ..... 2214
show ipv6 vrrp-extended ..... 2218
show issu errors ..... 2221
show issu sequence ..... 2222
show issu status ..... 2223
show keychain ..... 2225
show lag ..... 2227
show license ..... 2233
show license installed ..... 2235
show license node-locked ..... 2237
show license non-node-locked ..... 2239
show license unit ..... 2241
show link-error-disable ..... 2243
show link-keepalive ..... 2245
show link-oam info ..... 2247
show link-oam statistics ..... 2251
show lldp ..... 2255
show lldp local-info ..... 2257
show lldp neighbors ..... 2260
show lldp statistics ..... 2262
show local-userdb ..... 2264
show logging ..... 2266
show loop-detection resource ..... 2268
show loop-detection status ..... 2269
show loop-detect no-shutdown-status ..... 2270
show lrm-adapter ethernet ..... 2271
show l2protocol dot1q-tunnel ..... 2272
show mac-address ..... 2275
show mac-address cluster ..... 2277
show mac-address mdb ..... 2279
show mac-authentication configuration ..... 2280
show mac-authentication ip-acl ..... 2283
show mac-authentication sessions ..... 2285
show mac-authentication sessions detail ..... 2287
show mac-authentication statistics ..... 2289
show macsec statistics ..... 2291
show management traffic exclusion ..... 2295
show management-vrf ..... 2296
show spx pe-port-vlan-resources ..... 2297

show media....2299
show memory....2304
show memory task....2307
show metro-ring....2309
show mirror....2312
show module....2313
show monitor....2315
show mstp....2316
show mstp root-protect....2319
show notification mac-movement....2320
show notification-mac....2322
show ntp associations....2323
show ntp status....2326
show openflow....2328
show openflow controller....2330
show openflow flows....2331
show openflow groups....2333
show overlay-gateway....2335
show openflow interface....2338
show openflow meters....2340
show optic....2342
show optic-timer....2345
show packet-inerror-detect....2346
show pki....2347
show pod....2351
show port security....2353
show power-savings-statistics....2356
show priority-flow-control....2358
show protected-port....2359
show pvlan....2360
show pvstplus-protect-ports....2361
show qd-buffer-profile....2362
show qos egress-buffer-profile....2364
show qos ingress-buffer-profile....2366
show qos priority-to-pg....2367
show qos scheduler-profile....2369
show qos sflow-rate-limit....2372
show qos-internal-trunk-queue....2373
show qos-profiles....2374
show qos-tos....2375
show radius servers....2376
show rate-limit broadcast....2377
show rate-limit input....2378
show rate-limit output-shaping....2379
show rate-limit unknown-unicast....2380
show rear-module....2381
show relative-utilization....2382
show reserved-vlan-map....2383
show rmon....2384
show rmon statistics....2389

show rspan-vlan ..... 2392
show running ikev2 ..... 2393
show running interface ..... 2394
show running-config ..... 2396
show running-config interface ethernet ..... 2401
show running-config interface tunnel ..... 2402
show running-config interface ve ..... 2403
show running-config vlan ..... 2404
show scheduler-profile ..... 2405
show sflow ..... 2406
show snmp ..... 2408
show span ..... 2411
show span designated-protect ..... 2416
show spx ..... 2417
show spx connections ..... 2422
show spx csp ..... 2424
show spx debug ..... 2430
show spx mecid ..... 2433
show spx multicast cache ..... 2437
show spx multicast optimization ..... 2439
show spx multicast resource ..... 2441
show spx pe-port-vlan-resources ..... 2442
show spx zero-touch ipc ..... 2444
show spx zero-touch log ..... 2446
show spx zero-touch status ..... 2448
show spx-mon ..... 2449
show stack ..... 2453
show stack connection ..... 2456
show stack detail ..... 2458
show stack failover ..... 2460
show stack flash ..... 2461
show stack link-sync ..... 2462
show stack neighbors ..... 2463
show stack rel-ipc stats ..... 2464
show stack stack-ports ..... 2467
show startup-config (SPX) ..... 2468
show statistics ..... 2470
show statistics dos-attack ..... 2475
show statistics stack-ports ..... 2476
show statistics traffic-policy ..... 2477
show statistics tunnel ..... 2479
show stp-bpdu-guard ..... 2481
show stp-group ..... 2482
show stp-protect-ports ..... 2483
show symmetric-flow-control ..... 2484
show sz status ..... 2485
show tech-support ..... 2486
show telnet ..... 2498
show topology-group ..... 2500
show traffic-policy ..... 2502

show transmit-counter ... 2504
show users ... 2506
show version ... 2507
show vlan ... 2510
show vlan-group ... 2513
show voice-vlan ... 2514
show vrf ... 2515
show vsrp ... 2517
show vxlan tunnel ... 2520
show webauth ... 2521
show who ... 2525

**Commands Sn - Z ... 2527**

slow-start ... 2527
snmp-client ... 2528
snmp-server community ... 2529
snmp-server contact ... 2531
snmp-server disable ... 2532
snmp-server enable ... 2534
snmp-server enable mib ... 2535
snmp-server enable traps ... 2536
snmp-server enable traps holddown-time ... 2538
snmp-server enable traps mac-notification ... 2539
snmp-server engineid local ... 2540
snmp-server group ... 2542
snmp-server host ... 2544
snmp-server legacy ... 2546
snmp-server location ... 2547
snmp-server max-ifindex-per-module ... 2548
snmp-server preserve-statistics ... 2549
snmp-server pw-check ... 2550
snmp-server trap-source ... 2551
snmp-server user ... 2552
snmp-server view ... 2554
source-guard enable ... 2556
source-interface ... 2559
source-ip ... 2561
source-guard enable ... 2562
spanning-tree ... 2565
spanning-tree 802-1w ... 2567
spanning-tree 802-1w ethernet ... 2569
spanning-tree (Ethernet, LAG) ... 2571
spanning-tree designated-protect ... 2573
spanning-tree path-cost-method ... 2574
spanning-tree root-protect ... 2575
spanning-tree rstp ... 2576
speed-duplex ... 2577
spt-threshold ... 2580
spx allow-pe-movement ... 2582
spx cb-configure ... 2583
spx cb-enable ... 2585

spx interactive-setup....2586
spx pe-enable....2590
spx ping....2592
spx suggested-id....2594
spx unconfigure....2595
spx unit....2597
spx zero-touch-deny....2601
spx-lag....2602
spx-mon enable....2604
spx-port....2605
ssh....2607
ssh access-group....2609
ssm-enable....2610
stack disable....2612
stack enable....2613
stack mac....2614
stack secure-setup....2616
stack suggested-id....2617
stack suppress-warning....2618
stack switch-over....2619
stack unconfigure....2620
stack-port....2622
stack-trunk....2623
static-mac-address....2624
static-mac-ip-mapping....2626
store-and-forward....2627
stp-bpdu-guard....2629
stp-group....2630
stp-protect....2631
summary-address (OSPFv2)....2632
summary-address (OSPFv3)....2634
supportsave (SCP)....2636
suppress-acl-seq....2640
switch-over-active-role....2641
symmetric-flow-control enable....2642
symmetric-flow-control set....2643
symmetrical-flow-control enable....2645
system-max gre-tunnels....2647
system-max hw-traffic-conditioner....2648
system-max igmp-snoop-group-addr....2649
system-max igmp-snoop-mcache....2650
system-max ip-route....2651
system-max ip-route-default-vrf....2652
system-max ip-route-vrf....2653
system-max ip-vrf....2654
system-max ip6-route....2655
system-max ip6-route-default-vrf....2656
system-max ip6-route-vrf....2657
system-max ip-static-arp....2658
system-max ip-subnet-port....2659

system-max l3-interface ..... 2660
system-max mac ..... 2661
system-max mac-notification-buffer ..... 2662
system-max max-dhcp-snoop-entries ..... 2663
system-max max-ecmp ..... 2665
system-max max-ip-mac ..... 2667
system-max max-static-inspect-arp-entries ..... 2669
system-max mld-snoop-group-addr ..... 2671
system-max mld-snoop-mcache ..... 2672
system-max msdp-sa-cache ..... 2673
system-max pim-hw-mcache ..... 2674
system-max pim6-hw-mcache ..... 2675
system-max pms-global-pool ..... 2676
system-max rmon-entries ..... 2677
system-max spanning-tree ..... 2678
system-max view ..... 2679
system-max virtual-interface ..... 2680
system-max vlan ..... 2681
sz active-list ..... 2682
sz disable ..... 2684
sz disconnect ..... 2685
sz passive ..... 2686
sz query ..... 2688
sz registrar ..... 2689
sz registrar-list ..... 2690
sz registrar-query-restart ..... 2691
table-map ..... 2692
tacacs-server deadtime ..... 2694
tacacs-server enable ..... 2695
tacacs-server host ..... 2696
tacacs-server key ..... 2698
tacacs-server retransmit ..... 2699
tacacs-server timeout ..... 2700
tag-profile ..... 2701
tag-profile enable ..... 2702
tagged ethernet ..... 2703
tagged lag ..... 2704
telnet ..... 2705
telnet access-group ..... 2706
telnet client ..... 2707
telnet login-retries ..... 2708
telnet login-timeout ..... 2709
telnet server enable ..... 2710
telnet server suppress-reject-message ..... 2711
telnet strict-management-vrf ..... 2712
telnet timeout ..... 2714
temperature warning ..... 2715
terminal logging ..... 2716
terminal monitor ..... 2717
tftp client enable ..... 2718

tftp disable ..... 2719
tftp-server ..... 2720
tftp-server (IMAGE) ..... 2721
timers (BGP) ..... 2722
timers (OSPFv2) ..... 2723
timers (OSPFv3) ..... 2725
timers (RIP) ..... 2727
timers (RIPng) ..... 2728
timeout (EFM-OAM) ..... 2730
tolerance ..... 2731
topology-group ..... 2732
traceroute ..... 2733
track-port ..... 2735
track-port (VSRP) ..... 2737
traffic-policy count ..... 2739
traffic-policy rate-limit adaptive ..... 2740
traffic-policy rate-limit fixed ..... 2742
transform ..... 2744
trunk-threshold ..... 2745
trust dscp ..... 2747
trust-port ..... 2748
tunnel destination ..... 2749
tunnel mode gre ip ..... 2751
tunnel mode ipsec ..... 2752
tunnel mode ipv6ip ..... 2753
tunnel path-mtu-discovery ..... 2754
tunnel protection ipsec profile ..... 2755
tunnel source ..... 2756
tunnel tos ..... 2758
tunnel vrf ..... 2759
unknown-unicast limit (enable) ..... 2760
unknown-unicast limit (logging) ..... 2761
unmount disk0 ..... 2762
untagged ..... 2763
update-lag-name ..... 2764
update-time (BGP) ..... 2765
update-time (RIP) ..... 2766
use-radius-server ..... 2767
use-v2-checksum ..... 2768
use-vrrp-path (RIP) ..... 2769
username ..... 2770
username (Local database) ..... 2773
vendor-class ..... 2774
verify ..... 2775
version ..... 2777
violation ..... 2778
virtual-ip ..... 2780
virtual-port ..... 2781
vlan ..... 2782
vlan-config ..... 2784

vlan-group..... 2787
voice-vlan..... 2788
vrf..... 2789
vrf forwarding..... 2790
vsrp..... 2791
vsrp auth-type..... 2792
vsrp-aware..... 2793
web access-group..... 2795
web client..... 2796
web-management..... 2797
webauth..... 2800
webauth-redirect-address..... 2801
webpage custom-text..... 2802
webpage logo..... 2804
webpage terms..... 2806
wpad..... 2807
write terminal..... 2808
xwindow-manager..... 2810
zero-touch-enable..... 2811
zero-touch-ports..... 2813

# Preface


- Document Conventions....41
- Command Syntax Conventions....42
- Document Feedback....42
- Ruckus Product Documentation Resources....42
- Online Training Resources....43
- Contacting Ruckus Customer Services and Support....43


# Document Conventions

The following table lists the text conventions that are used throughout this guide.

**TABLE 1 Text Conventions**

| Convention | Description | Example |
|---|---|---|
| `monospace` | Identifies command syntax examples | `device(config)# interface ethernet 1/1/6` |
| **bold** | User interface (UI) components such as screen or page names, keyboard keys, software buttons, and field names | On the **Start** menu, click **All Programs**. |
| *italics* | Publication titles | Refer to the *Ruckus Small Cell Release Notes* for more information. |

## Notes, Cautions, and Warnings

Notes, cautions, and warning statements may be used in this document. They are listed in the order of increasing severity of potential hazards.

**NOTE**
A NOTE provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

**ATTENTION**
An ATTENTION statement indicates some information that you must read before continuing with the current action or task.

**CAUTION**
**A CAUTION statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.**

**DANGER**
***A DANGER statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.***

# Command Syntax Conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

| Convention | Description |
|---|---|
| **bold** text | Identifies command names, keywords, and command options. |
| *italic* text | Identifies a variable. |
| [ ] | Syntax components displayed within square brackets are optional. |
| | Default responses to system prompts are enclosed in square brackets. |
| { **x** \| **y** \| **z** } | A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options. |
| **x** \| **y** | A vertical bar separates mutually exclusive elements. |
| < > | Nonprinting characters, for example, passwords, are enclosed in angle brackets. |
| ... | Repeat the previous element, for example, *member*[*member*...]. |
| \ | Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash. |

# Document Feedback

Ruckus is interested in improving its documentation and welcomes your comments and suggestions.

You can email your comments to Ruckus at #Ruckus-Docs@commscope.com.

When contacting us, include the following information:

- Document title and release number
- Document part number (on the cover page)
- Page number (if appropriate)

For example:

- Ruckus SmartZone Upgrade Guide, Release 5.0
- Part number: 800-71850-001 Rev A
- Page 7

# Ruckus Product Documentation Resources

Visit the Ruckus website to locate related documentation for your product and additional Ruckus resources.

Release Notes and other user documentation are available at https://support.ruckuswireless.com/documents. You can locate the documentation by product or perform a text search. Access to Release Notes requires an active support contract and a Ruckus Support Portal user account. Other technical documentation content is available without logging in to the Ruckus Support Portal.

White papers, data sheets, and other product documentation are available at https://www.ruckuswireless.com.

# Online Training Resources

To access a variety of online Ruckus training modules, including free introductory courses to wireless networking essentials, site surveys, and Ruckus products, visit the Ruckus Training Portal at https://training.ruckuswireless.com.

# Contacting Ruckus Customer Services and Support

The Customer Services and Support (CSS) organization is available to provide assistance to customers with active warranties on their Ruckus products, and customers and partners with active support contracts.

For product support information and details on contacting the Support Team, go directly to the Ruckus Support Portal using https://support.ruckuswireless.com, or go to https://www.ruckuswireless.com and select **Support**.

## What Support Do I Need?

Technical issues are usually described in terms of priority (or severity). To determine if you need to call and open a case or access the self-service resources, use the following criteria:

- Priority 1 (P1)—Critical. Network or service is down and business is impacted. No known workaround. Go to the **Open a Case** section.
- Priority 2 (P2)—High. Network or service is impacted, but not down. Business impact may be high. Workaround may be available. Go to the **Open a Case** section.
- Priority 3 (P3)—Medium. Network or service is moderately impacted, but most business remains functional. Go to the **Self-Service Resources** section.
- Priority 4 (P4)—Low. Requests for information, product documentation, or product enhancements. Go to the **Self-Service Resources** section.

## Open a Case

When your entire network is down (P1), or severely impacted (P2), call the appropriate telephone number listed below to get help:

- Continental United States: 1-855-782-5871
- Canada: 1-855-782-5871
- Europe, Middle East, Africa, Central and South America, and Asia Pacific, toll-free numbers are available at https://support.ruckuswireless.com/contact-us and Live Chat is also available.
- Worldwide toll number for our support organization. Phone charges will apply: +1-650-265-0903

We suggest that you keep a physical note of the appropriate support number in case you have an entire network outage.

## Self-Service Resources

The Ruckus Support Portal at https://support.ruckuswireless.com offers a number of tools to help you to research and resolve problems with your Ruckus products, including:

- Technical Documentation—https://support.ruckuswireless.com/documents

- Community Forums—https://forums.ruckuswireless.com/ruckuswireless/categories
- Knowledge Base Articles—https://support.ruckuswireless.com/answers
- Software Downloads and Release Notes—https://support.ruckuswireless.com/#products_grid
- Security Bulletins—https://support.ruckuswireless.com/security

Using these resources will help you to resolve some issues, and will provide TAC with additional data from your troubleshooting analysis if you still require assistance through a support case or RMA. If you still require help, open and manage your case at https://support.ruckuswireless.com/case_management.

# About This Document


- What's new in this document....45
- Supported hardware....48


# What's new in this document

Information has been added or updated to reflect new FastIron features or enhancements to existing FastIron features.

For commands introduced since Release 08.0.01, a history table is included with each command to provide details about the modifications to that command. For commands introduced prior to Release 08.0.01, a history table is not provided, unless the command has been modified in recent releases.

**NOTE**
In addition to commands that are new or modified for this release, commands for existing FastIron features may have been added that were previously described only in FastIron configuration guides.

## New and modified commands for FastIron 08.0.80c

The following commands have been introduced or modified this release.

- **show sz status**
- **sz passive**
- **sz registrar**
- **sz registrar-list**
- **sz registrar-restart**

## New commands for FastIron 08.0.80

The following commands have been added (new for this release).

- **auth allow-tagged enable**
- **auth auth-mode**
- **auth-mode**
- **clear authentication sessions**
- **clear authentication statistics**
- **copy https flash**
- **copy https startup-config**
- **copy running-config https**
- **copy startup-config https**
- **dot1x macauth-override**
- **erase system factory-default**
- **inline power couple-datalink**
- **ip dhcp snooping relay information disable**

- **ipv6 nd ra-domain-name**
- **ipv6 nd ra-dns-server**
- **ipv6 nd suppress-ra address**
- **license delete perpetual**
- **license set serial-number**
- **mac-authentication dot1x-disable**
- **max-vlans-per-pe-port** (SPX)
- **rspan destination**
- **rspan source**
- **rspan-vlan**
- **show authentication acls**
- **show authentication configuration**
- **show authentication sessions**
- **show authentication statistics**
- **show boot-monitor**
- **show rspan-vlan**
- **show spx pe-port-vlan-resources**
- **sz active-list**
- **sz disconnect**
- **sz disable**
- **sz query**

## Modified commands for FastIron 08.0.80

The following commands have been modified in this release.

- **authentication max-sessions**
- **copy scp flash**
- **copy tftp flash**
- **copy tftp system-manifest**
- **crypto key client generate**
- **crypto key generate**
- **ip arp inspection vlan**
- **ip dhcp snooping vlan**
- **ipv6 dhcp6 snooping vlan**
- **ipv6 neighbor inspection vlan**
- **license delete unit**
- **license install perpetual**
- **logging buffered**
- **radius-server host**
- **show boot-monitor**

- **show chassis**
- **show ip dhcp-client options**
- **show ip dhcp-server address-pool**
- **show license**
- **show license installed**
- **show license node-locked**
- **show license non-node-locked**
- **show license unit**
- **show logging**
- **show running-config vlan**
- **show version**
- **source-guard enable**
- **system-max max-dhcp-snoop-entries**
- **system-max max-static-inspect-arp-entries**
- **tagged ethernet**

# Deprecated commands for FastIron 08.0.80

The following commands have been deprecated in this release.

- **access-list (standard numbered)**
- **access-list enable accounting**
- **access-list remark**
- **dual-mode**
- **lldp-pass-through** (Flexible authentication)
- **max-vlan** (SPX)

Beginning with the FastIron 08.0.61 release, Layer 3 features for the Ruckus ICX 7150 are supported. The following Layer 3 features are not supported for the Ruckus ICX 7150, and this has been noted where applicable throughout this guide:

- BGP4
- BGP4+
- Multi-VRF
- Tunnels
- uRPF

# Other enhancements for FastIron 08.0.80

**TABLE 2 Other enhancements in FastIron release 08.0.80**

| Feature | Description | Location |
|---|---|---|
| Tab-based auto-completion of CLI nodes | If there are more than one command or keyword associated with the characters typed, pressing Tab auto-fills the nodes up to the last common matching character among all the nodes so that | Command completion on page 50 |

**TABLE 2 Other enhancements in FastIron release 08.0.80 (continued)**

| Feature | Description | Location |
|---|---|---|
| | typing a single character allows you to auto-fill to complete the keyword. | |
| Updates to address defects | Minor updates on commands throughout to address defects. | All chapters. |
| Minor editorial updates | Minor editorial updates were made throughout the Command Reference. | All chapters. |

# Supported hardware

This guide supports the following Ruckus products:

- Ruckus ICX 7750 Series
- Ruckus ICX 7650 Series
- Ruckus ICX 7450 Series
- Ruckus ICX 7250 Series
- Ruckus ICX 7150 Series

For information about what models and modules these devices support, see the hardware installation guide for the specific product family.

# Using the FastIron Command-Line Interface


- Accessing the CLI..... 49
- Searching and filtering command output..... 52
- Creating an alias for a CLI command..... 56


# Accessing the CLI

Once an IP address is assigned to a Ruckus device running Layer 2 software or to an interface on the Ruckus device running Layer 3 software, you can access the CLI either through a direct serial connection or through a local or remote Telnet session.

You can initiate a local Telnet or SNMP or SSH connection by attaching a cable to a port and specifying the assigned management station IP address.

## Command configuration modes

The Ruckus CLI uses an industry-standard hierarchical shell familiar to Ethernet/IP networking administrators. You can use one of three major command modes to enter commands and access sub-configuration modes on the device.

### *User EXEC mode*

User EXEC mode is the default mode for the device; it supports the lowest level of user permissions. In this mode, you can execute basic commands such as **ping** and **traceroute**, but only a subset of clear, show, and debug commands can be entered in this mode. The following example shows the User EXEC prompt after login. The **enable** command enters privileged EXEC mode.

```
device> enable
device#
```

### *Privileged EXEC mode*

Privileged EXEC mode supports all clear, show, and debug commands. In addition, you can enter some configuration commands that do not make changes to the system configuration. The following example shows the privileged EXEC prompt. At this prompt, you issue the **configure terminal** command to enter global configuration mode.

```
device# configure terminal
device(config)#
```

### *Global configuration mode*

Global configuration mode supports commands that can change the device configuration. For any changes to be persistent, you must save the system configuration before rebooting the device. The global configuration mode provides access to sub-configuration modes for individual interfaces, VLANs, routing protocols, and other configuration areas. The following example shows how you access the interface sub-configuration mode by issuing the **interface** command with a specified interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)#
```

# Command help

You can display commands and syntax information in any mode and from any point in the command hierarchy.

Enter a question mark (?) or a tab in any command mode to display the list of commands available in that mode.

```
device(config)#?
  aaa                       Define authentication method list
  access-list               Define Access Control List (ACL)
  aggregated-vlan           Support for larger Ethernet frames up to 1536 bytes
  alias                     Configure alias or display configured alias
  all-client                Restrict all remote management to a host
  arp                       Enter a static IP ARP entry
  arp-internal-priority     Set packet priority
  arp-subnet-only           Only learn ARP in the subnet of this device
  authentication            Configure flexible authentication
  banner                    Define a login banner
  batch                     Define a group of commands
  boot                      Set system boot options
  (output truncated)
```

To display a list of commands that start with a specified character, type the character followed by a question mark (?) or a tab.

```
device(config)#e?
  enable                    Password, page-mode and other options
  end                       End Configuration level and go to Privileged level
  errdisable                Set Error Disable Attributions
  exit                      Exit current level
  extern-config-file        Extern configuration file
```

To display keywords and arguments associated with a command, enter the command followed by a question mark (?) or a tab.

```
deviceh(config)#qos ?
  egress-buffer-profile   User defined QoS egress profile
  mechanism               Change mechanism
  name                    Change name
  profile                 Change bandwidth allocation
  scheduler-profile       User defined QoS profile
  tagged-priority         Change tagged frame priority to profile mapping
```

# Command completion

Command completion allows you to run a command by entering a partial string.

To complete the spelling of commands or keywords automatically, begin typing the command or keyword and then press Tab. For example, entering **conf t** in privileged EXEC mode auto-completes the keyword and executes the **configure terminal** command.

```
device# conf t
  terminal   Configure thru terminal
device# conf terminal
device(config)#
```

In releases prior to 08.0.80, if there are two or more matching nodes, the node will not be auto-filled when Tab is pressed. You must type all the characters in the node to differentiate among the nodes. In the below example "dh" is matched with multiple nodes. Pressing Tab does not auto-complete the keyword but has to manually type all the characters in the node.

```
device(config)#ip dh
  dhcp                          Set DHCP option
  dhcp-client                   DHCP client options
  dhcp-server                   DHCP Server
  dhcp-valid-check              Check DHCP offer packet for NULL client addr
```

In 08.0.80 release onwards, if there are more than one command or keyword associated with the characters typed, pressing Tab auto-fills the nodes up to the last common matching character among all the nodes so that typing a single character allows you to auto-fill to complete the keyword.

```
device(config)# ip dh<and press Tab>

device(config)# ip dhcp
 dhcp                          Set DHCP option
 dhcp-client                   DHCP client options
 dhcp-server                   DHCP Server
 dhcp-valid-check              Check DHCP offer packet for NULL client addr
```

In the example shown above, since dhcp is the common word among the 4 options, issuing Tab will autofill "dh" to "dhcp". The CLI displays all choices matching the characters. Type another character to differentiate among the nodes and utilize the Tab-based command completion thus improving the usability.

```
device(config)# ip dhcp-c<and press Tab>

device(config)# ip dhcp-client
 auto-update                   Enable the DHCP client auto-update
 disable                       disable DHCP client globally on router
```

If you enter an invalid command or partial string that cannot be completed, an error message is displayed.

```
device(config)#shw
Unrecognized command
device(config)#shw
```

# Scroll control

By default, the CLI uses a page mode to paginate displays that are longer than 23 lines. The maximum number of lines per page is 23 (line 24 is reserved for printing). Displays that are longer than 23 lines are automatically segmented into pages with 23 lines per page.

If you use the question mark (?) to display a listing of available options in a given mode, the display stops at each 23 line increment and lists your choices for continuing the display.

```
aaa
all-client
appletalk
arp
boot
some lines omitted for brevity...

ipx
lock-address
logging
mac
--More--, next page: Space, next line:
Return key, quit: Control-c
```

Use one of the following scrolling options to display additional information:

- Press the **Space bar** to display the next page (one screen at a time).
- Press the **Return** or **Enter** key to display the next line (one line at a time).
- Press **Ctrl+C** or **Ctrl+Q** to cancel the display.
- Use ths **skip** command in privileged EXEC mode to disable page display mode. Use the **page** command to re-enable page display mode

The following example toggles between page display modes.

```
device# skip
Disable page display mode
device# page
Enable page display mode
```

## Line editing commands

The CLI supports the following line editing commands. To enter a line-editing command, use the CTRL+key combination for the command by pressing and holding the CTRL key, then pressing the letter associated with the command.

**TABLE 3** CLI line editing commands

| Ctrl+Key combination | Description |
|---|---|
| Ctrl+A | Moves to the first character on the command line. |
| Ctrl+B | Moves the cursor back one character. |
| Ctrl+C | Escapes and terminates command prompts and ongoing tasks (such as lengthy displays), and displays a fresh command prompt. |
| Ctrl+D | Deletes the character at the cursor. |
| Ctrl+E | Moves to the end of the current command line. |
| Ctrl+F | Moves the cursor forward one character. |
| Ctrl+K | Deletes all characters from the cursor to the end of the command line. |
| Ctrl+L; Ctrl+R | Repeats the current command line on a new line. |
| Ctrl+N | Enters the next command line in the history buffer. |
| Ctrl+P | Enters the previous command line in the history buffer. |
| Ctrl+U; Ctrl+X | Deletes all characters from the cursor to the beginning of the command line. |
| Ctrl+W | Deletes the last word you typed. |
| Ctrl+Z | Moves from any CONFIG level of the CLI to the Privileged EXEC level; at the Privileged EXEC level, moves to the User EXEC level. |

# Searching and filtering command output

You can filter the output from **show** commands at the --More-- prompt. You can search for characters strings, or you can construct complex regular expressions to filter the output.

## Searching and filtering output at the --More-- prompt

The --More-- prompt displays when output extends beyond a single page. At this prompt, you can press the Space bar to display the next page, the Return or Enter key to display the next line, or Ctrl+C or Q to cancel the display. In addition, you can search and filter output from this prompt.

At the --More-- prompt, enter a forward slash ( / ) followed by a search string. The Ruckus device displays output starting from the first line that contains the search string as shown in the following example. The search feature is similar to the **begin** option for **show** commands.

```
--More--, next page: Space, next line: Return key, quit: Control-c
/telnet
```

The results of the search are displayed.

```
searching...
  telnet              Telnet by name or IP address
  temperature         temperature sensor commands
  terminal            display syslog
  traceroute          TraceRoute to IP node
  undebug             Disable debugging functions (see also 'debug')
  undelete            Undelete flash card files
  whois               WHOIS lookup
  write               Write running configuration to flash or terminal
```

To display lines containing only a specified search string (similar ) press the plus key (+) at the --More-- prompt followed by a search string. This option is similar to the **include** option supported with **show** commands.

```
--More--, next page: Space, next line: Return key, quit: Control-c
+telnet
```

The filtered results are displayed.

```
filtering...
  telnet              Telnet by name or IP address
```

To display lines that do not contain a specified search string, press the minus key (-) at the --More-- prompt followed by a search string. This option is similar to the **exclude** option supported with **show** commands.

```
--More--, next page: Space, next line: Return key, quit: Control-c
-telnet
```

The filtered results are displayed.

```
filtering...
  temperature         temperature sensor commands
  terminal            display syslog
  traceroute          TraceRoute to IP node
  undebug             Disable debugging functions (see also 'debug')
  undelete            Undelete flash card files
  whois               WHOIS lookup
  write               Write running configuration to flash or terminal
```

As with the commands for filtering output from **show** commands, the search string is a regular expression consisting of a single character or string of characters. You can use special characters to construct complex regular expressions. See the next section for information on special characters used with regular expressions.

# Searching and filtering show command output

You can filter output from **show** commands to display lines containing a specified string, lines that do not contain a specified string, or output starting with a line containing a specified string. The search string is a regular expression consisting of a single character or a string of characters. You can use special characters to construct complex regular expressions.

## *Using special characters to construct complex regular expressions*

Special characters allow you to construct complex regular expressions to filter output from **show** commands. You can use a regular expression to specify a single character or multiple characters as a search string. In addition, you can include special characters that influence the way the software matches the output against the search string. Supported special characters are listed in the following table.

**TABLE 4 Special characters for regular expressions**

| Character | Operation |
|---|---|
| . | The period matches on any single character, including a blank space.<br>For example, the following regular expression matches "aaz", "abz", "acz", and so on, but not just "az":<br>a.z |
| * | The asterisk matches on zero or more sequential instances of a pattern.<br>For example, the following regular expression matches output that contains the string "abc", followed by zero or more Xs:<br>abcX* |
| + | The plus sign matches on one or more sequential instances of a pattern.<br>For example, the following regular expression matches output that contains "de", followed by a sequence of "g"s, such as "deg", "degg", "deggg", and so on:<br>deg+ |
| ? | The question mark matches on zero occurrences or one occurrence of a pattern.<br>For example, the following regular expression matches output that contains "dg" or "deg":<br>de?g<br>**NOTE**<br>Normally when you type a question mark, the CLI lists the commands or options at that CLI level that begin with the character or string you entered. However, if you enter Ctrl +V and then type a question mark, the question mark is inserted into the command line, allowing you to use it as part of a regular expression. |
| ^ | A caret (when not used within brackets) matches on the beginning of an input string.<br>For example, the following regular expression matches output that begins with "deg":<br>^deg |
| $ | A dollar sign matches on the end of an input string.<br>For example, the following regular expression matches output that ends with "deg":<br>deg$ |
| _ | An underscore matches on one or more of the following:<br>• , (comma)<br>• { (left curly brace)<br>• } (right curly brace)<br>• ( (left parenthesis)<br>• ) (right parenthesis)<br>• The beginning of the input string<br>• The end of the input string<br>• A blank space |

**TABLE 4 Special characters for regular expressions (continued)**

| Character | Operation |
|---|---|
| | For example, the following regular expression matches on "100" but not on "1002", "2100", and so on.<br>_100_ |
| [ ] | Square brackets enclose a range of single-character patterns.<br>For example, the following regular expression matches output that contains "1", "2", "3", "4", or "5":<br>[1-5]<br>You can use the following expression symbols within the brackets. These symbols are allowed only inside the brackets.<br>• ^ - The caret matches on any characters except the ones in the brackets. For example, the following regular expression matches output that does not contain "1", "2", "3", "4", or "5":[^1-5]<br>• - The hyphen separates the beginning and ending of a range of characters. A match occurs if any of the characters within the range is present. See the example above. |
| \| | A vertical bar separates two alternative values or sets of values. The output can match one or the other value.<br>For example, the following regular expression matches output that contains either "abc" or "defg":<br>abc\|defg |
| ( ) | Parentheses allow you to create complex expressions.<br>For example, the following complex expression matches on "abc", "abcabc", or "defg", but not on "abcdefgdefg":<br>((abc)+)\|((defg)?) |

If you want to filter for a special character instead of using the special character as described in the table above, enter a backslash ( \ ) before the character. For example, to filter on output containing an asterisk, enter the asterisk portion of the regular expression as "*".

```
device#show ip route bgp | include \*
```

## *Displaying lines containing a specified string*

The following command filters the output of the **show interface** command for port 1/3/11 to display only lines containing the word "Internet". This command can be used to display the IP address of the interface.

```
device#show interface e 1/3/11 | include Internet
  Internet address is 10.168.1.11/24, MTU 1518 bytes, encapsulation ethernet
```

**Syntax:** *show-command* | **include** *regular-expression*

**NOTE**
The vertical bar ( **|** ) is part of the command.

Note that the regular expression specified as the search string is case sensitive. In the example above, a search string of "Internet" would match the line containing the IP address, but a search string of "internet" would not.

### *Displaying lines that do not contain a specified string*

The following command filters the output of the **show who** command to display only the lines that do not contain the word "closed". This command can be used to display open connections to the Ruckus device.

```
device#show who | exclude closed
Console connections:
        established
        you are connecting to this session
        2 seconds in idle
Telnet connections (inbound):
 1      established, client ip address 10.168.9.37
        27 seconds in idle
Telnet connection (outbound):
SSH connections:
```

**Syntax:** *show-command* | **exclude** *regular-expression*

### *Displaying lines starting with a specified string*

The following command filters the output of the **show who** command to display output starting with the first line that contains the word "SSH". This command can be used to display information about SSH connections to the Ruckus device.

```
device#show who | begin SSH
SSH connections:
 1      established, client ip address 10.168.9.210
        7 seconds in idle
 2      closed
 3      closed
 4      closed
 5      closed
```

**Syntax:** *show-command* | **begin** *regular-expression*

# Creating an alias for a CLI command

An alias serves as a shorthand version of a longer CLI command. For example, you can create an alias called *shoro* for the **show ip route** command. You can then enter te *shoro* alias at the command prompt and the **show ip route** command is issued.

To create an alias called *shoro* for the CLI command **show ip route**, enter the **alias shoro = show ip route** command.

```
device(config)# alias shoro = show ip route
```

**Syntax: [no] alias** *alias-name* **=** *cli-command*

The *alias-name* must be a single word, without spaces.

After the alias is configured, entering *shoro* in the privileged EXEC mode or in the global configuration mode issues the **show ip route** command.

Enter the command **copy running-config** with the appropriate parameters to create an alias called *wrsbc*.

```
device(config)#alias wrsbc = copy running-config tftp 10.10.10.10 test.cfg
```

To remove the *wrsbc* alias from the configuration, enter one of the following commands.

```
device(config)#no alias wrsbc
```

or

```
device(config)#unalias wrsbc
```

**Syntax: unalias** *alias-name*

The specified *alias-name* must be the name of an alias already configured on the Ruckus device.

To display the aliases currently configured on the Ruckus device, enter the following command in the Privileged EXEC mode or in the global configuration mode.

```
device# alias
          wrsbc      copy running-config tftp 10.10.10.10 test.cfg
          shoro       show ip route
```

**Syntax: alias**

# Configuration notes for creating a command alias

The following configuration notes apply to this feature:

- You cannot include additional parameters with the alias at the command prompt. For example, after you create the *shoro* alias, *shoro bgp* would not be a valid command.
- If configured on the Ruckus device, authentication, authorization, and accounting is performed on the actual command, not on the alias for the command.
- To save an alias definition to the startup-config file, use the **write memory** command.

# Commands A and B

## 100-fx

Enables 100Base-FX on chassis-based and stackable devices.

### Syntax

**100-fx**

**no 100-fx**

### Command Default

100Base-FX is not enabled after installation.

### Modes

Interface configuration mode

### Usage Guidelines

After you physically install a 100Base-FX transceiver, you must use this command to enable 100Base-FX support on the device.

FastIron ICX devices support the following types of SFPs for 100BaseFX:

- *Multimode SFP*—maximum distance is 2 kilometers
- *Long Reach (LR)*—maximum distance is 40 kilometers
- *Intermediate Reach (IR)* —maximum distance is 15 kilometers

For information about supported SFP and SFP+ transceivers on FastIron devices, refer to the *Ruckus Optics Family Datasheet* on the Ruckus website.

**NOTE**
You must disable 100Base-FX support before inserting a different type of module in the same port. Otherwise, the device will not recognize traffic traversing the port.

The **no** form of the command disables 100Base-FX support.

### Examples

The following example enables support for 100Base-FX on a fiber port.

```
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# 100-fx
```

# 100-tx

Configures a 1000Base-TX small form-factor pluggable (SFP) transceiver to operate at a speed of 100 Mbps.

## Syntax

**100-tx**

**no 100-tx**

## Command Default

1000Base-TX SFP transceiver is not configured to operate at a speed of 100 Mbps.

## Modes

Interface configuration mode

## Usage Guidelines

This command requires autonegotiation to be enabled on the other end of the link.

Although combo ports (ports 1 to 4) on Hybrid Fiber (HF) models support the 1000Base-TX SFP, they cannot be configured to operate at 100 Mbps. The 100-Mbps operating speed is supported only with noncombo ports (ports 5 to 24).

1000Base-TX modules must be configured individually, one interface at a time. 1000Base-TX modules do not support digital optical monitoring. Hotswap is supported for this module when it is configured in 100M mode.

The **no** form of the command disables 1000Base-TX SFP support.

## Examples

The following example configures a 1000Base-TX SFP transceiver to operate at 100 Mbps.

```
device(config)# interface ethernet 1/5/1
device(config-if-e1000-1/5/1)# 100-tx
```

# aaa accounting commands

Configures the AAA accounting configuration parameters for EXEC commands.

## Syntax

**aaa accounting commands** *privilege-level* **default start-stop radius** [ **tacacs+** ] [ **none** ]

**no aaa accounting commands** *privilege-level* **default start-stop radius** [ **tacacs+** ] [ **none** ]

**aaa accounting commands** *privilege-level* **default start-stop tacacs+** [ **radius** ] [ **none** ]

**no aaa accounting commands** *privilege-level* **default start-stop tacacs+** [ **radius** ] [ **none** ]

**aaa accounting commands** *privilege-level* **default start-stop none**

**no aaa accounting commands** *privilege-level* **default start-stop none**

## Command Default

AAA accounting is disabled.

## Parameters

*privilege-level*

Configures the device to perform AAA accounting for the commands available at the specified privilege level. Valid values are 0 (Super User level - all commands), 4 (Port Configuration level - port-config and read-only commands), and 5 (Read Only level - read-only commands).

**default**

Configures the default named list.

**start-stop**

Configures to send an Accounting Start packet to the AAA accounting server when you enter a command, and an Accounting Stop packet when the service provided by the command is completed.

**radius**

Configures RADIUS accounting.

**tacacs+**

Configures TACACS+ accounting.

**none**

Disables accounting. This is equivalent to using the **no** form of the command.

## Modes

Global configuration mode

## Usage Guidelines

You can configure AAA accounting for CLI commands by specifying a privilege level whose commands require accounting.

**NOTE**
If authorization is enabled, and the command requires authorization, then authorization is performed before accounting takes place. If authorization fails for the command, no accounting takes place.

You can configure RADIUS, TACACS+, and None as accounting methods. If the configured primary accounting fails due to an error, the device tries the backup accounting methods in the order they are configured.

The **no** form of the command disables accounting.

# Examples

The following example shows how to configure the ICX device to perform RADIUS accounting for the commands available at the Super User privilege level (that is, all commands on the device).

```
device(config)# aaa accounting commands 0 default start-stop radius
```

The following example shows how to configure the ICX device to perform TACACS+ accounting for the commands available at the Read-only level (that is, read-only commands). The command also configures TACACS+ as the primary accounting followed by RADIUS.

```
device(config)# aaa accounting commands 5 default start-stop tacacs+ radius
```

# aaa accounting dot1x

Enables 802.1X accounting.

## Syntax

**aaa accounting dot1x default start-stop radius** [ **none** ]

**no aaa accounting dot1x default start-stop radius** [ **none** ]

**aaa accounting dot1x default start-stop none**

**no aaa accounting dot1x default start-stop none**

## Command Default

AAA accounting is disabled.

## Parameters

**default**

Configures the default named list.

**start-stop**

Configures to sent an Accounting Start packet is sent to the RADIUS accounting server when 802.1x session is enabled, and an Accounting Stop packet is sent when the service provided by the command is completed.

**radius**

Configures RADIUS accounting.

**none**

Disables accounting. The client is automatically authenticated without the device using information supplied by the client.

## Modes

Global configuration mode

## Usage Guidelines

You can configure both RADIUS and None as accounting methods. If the configured primary accounting fails due to an error, the device tries the backup accounting methods in the order they are configured.

**NOTE**
If authorization is enabled, and the command requires authorization, then authorization is performed before accounting takes place. If authorization fails for the command, no accounting takes place.

The **no** form of the command disables accounting.

# Examples

The following example shows how to enable 802.1x accounting.

```
device(config)# aaa accounting dot1x default start-stop radius
```

The following example shows how to enable 802.1x accounting and configure RADIUS as the primary accounting method. If the configured primary RADIUS accounting fails due to an error, the device tried the backup accounting method "none", that is, accounting will be disabled.

```
device(config)# aaa accounting dot1x default start-stop radius none
```

# aaa accounting exec

Configures the AAA accounting configuration parameters for SSH and Telnet access.

## Syntax

**aaa accounting exec default start-stop radius** [ **tacacs+** ] [ **none** ]

**no aaa accounting exec default start-stop radius** [ **tacacs+** ] [ **none** ]

**aaa accounting exec default start-stop tacacs+** [ **radius** ] [ **none** ]

**no aaa accounting exec default start-stop tacacs+** [ **radius** ] [ **none** ]

**aaa accounting exec default start-stop none**

**no aaa accounting exec default start-stop none**

## Command Default

AAA accounting is disabled.

## Parameters

**default**
Configures the default named list.

**start-stop**
Configures to send an Accounting Start packet to the AAA accounting server when an authenticated user establishes a Telnet or SSH session on the ICX device, and an Accounting Stop packet when the user logs out.

**radius**
Configures RADIUS accounting.

**tacacs+**
Configures TACACS+ accounting.

**none**
Disables accounting.

## Modes

Global configuration mode

## Usage Guidelines

You can configure RADIUS, TACACS+, and None as accounting methods. If the configured primary accounting fails due to an error, the device tries the backup accounting methods in the order they are configured.

**NOTE**
If authorization is enabled, and the command requires authorization, then authorization is performed before accounting takes place. If authorization fails for the command, no accounting takes place.

The **no** form of the command disables accounting.

# Examples

The following example shows how to configure the ICX device to perform RADIUS accounting for Telnet or SSH access.

```
device(config)# aaa accounting exec default start-stop radius
```

The following example shows how to configure the ICX device to perform TACACS+ accounting for Telnet or SSH access and to specify the order of accounting preference.

```
device(config)# aaa accounting exec default start-stop tacacs+ radius none
```

# aaa accounting mac-auth

Enables or disables RADIUS accounting for MAC authentication sessions.

## Syntax

**aaa accounting mac-auth default start-stop radius** [ **none** ]

**no aaa accounting mac-auth default start-stop radius** [ **none** ]

**aaa accounting mac-auth default start-stop none**

**no aaa accounting mac-auth default start-stop none**

## Command Default

AAA accounting is disabled.

## Parameters

**default**

Configures the default named list.

**start-stop**

Configures an accounting start packet to be sent to the RADIUS accounting server when a MAC authentication session is enabled and an accounting stop packet to be sent when the service provided by the command is completed.

**radius**

Configures RADIUS accounting.

**none**

Disables accounting. The client is automatically authenticated without the device using information supplied by the client.

## Modes

Global configuration mode

## Usage Guidelines

You can configure both RADIUS and None as accounting methods. If the configured primary accounting fails due to an error, the device tries the backup accounting methods in the order in which they are configured.

**NOTE**
If authorization is enabled, and the command requires authorization, then authorization is performed before accounting occurs. If authorization fails for the command, no accounting takes place.

The **no** form of the command disables accounting.

# Examples

The following example shows how to enable accounting for MAC authentication sessions.

```
device(config)# aaa accounting mac-auth default start-stop radius
```

The following example shows how to enable accounting for MAC authentication and configure RADIUS as the primary accounting method. If the configured primary RADIUS accounting fails due to an error, the device tries the backup accounting method **none**, that is, accounting is disabled.

```
device(config)# aaa accounting mac-auth default start-stop radius none
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# aaa accounting system

Configures AAA accounting to record when system events occur on the device.

## Syntax

**aaa accounting system default start-stop radius** [ **tacacs+** ] [ **none** ]

**no aaa accounting system default start-stop radius** [ **tacacs+** ] [ **none** ]

**aaa accounting system default start-stop tacacs+** [ **radius** ] [ **none** ]

**no aaa accounting system default start-stop tacacs+** [ **radius** ] [ **none** ]

**aaa accounting system default start-stop none**

**no aaa accounting system default start-stop none**

## Command Default

AAA accounting is disabled.

## Parameters

**default**
Configures the default named list.

**start-stop**
Configures to send an Accounting Start packet to be sent to the AAA accounting server when a system event occurs, and an Accounting Stop packet to be sent when the system event is completed.

**radius**
Configures RADIUS accounting.

**tacacs+**
Configures TACACS+ accounting.

**none**
Disables accounting.

## Modes

Global configuration mode

## Usage Guidelines

You can configure RADIUS, TACACS+, and None as accounting methods. If the configured primary accounting fails due to an error, the device tries the backup accounting methods in the order they are configured.

The **no** form of the command disables accounting.

# Examples

The following example shows how to configure the ICX device to perform RADIUS accounting to record when a system event occurs.

```
device(config)# aaa accounting system default start-stop radius
```

The following example shows how to configure the device to perform TACACS+ accounting to record when a system event occurs and to specify RADIUS and None as the backup accounting methods.

```
device(config)# aaa accounting system default start-stop tacacs+ radius none
```

# aaa authentication dot1x

Enables 802.1X and MAC authentication.

## Syntax

**aaa authentication dot1x default radius** [ **none** ]

**no aaa authentication dot1x default radius** [ **none** ]

**aaa authentication dot1x default none**

**no aaa authentication dot1x default none**

## Command Default

AAA authentication is disabled.

## Parameters

**default**

Configures the default named list.

**radius**

Configures RADIUS authentication.

**none**

Disables authentication. The client is automatically authenticated by other means, without the device using information supplied by the client.

## Modes

Global configuration mode

## Usage Guidelines

To use 802.1X and MAC authentication, you must specify an authentication method to be used to authenticate clients. RADIUS authentication with 802.1X authentication is supported. To use RADIUS authentication with 802.1X authentication, you create an authentication method list for 802.1X and specify RADIUS as an authentication method, and then configure communication between the device and the RADIUS server.

If you specify both **RADIUS** and **none**, ensure **RADIUS** comes before **none** when the command is used.

You can configure the RADIUS and None as authentication methods. If the configured primary authentication fails due to an error, the device tries the backup authentication methods in the order they are configured.

Use the **aaa authentication dot1x default radius** command for both MAC authentication and 802.1X authentication.

The **no** form of the command disables authentication.

# Examples

The following example enables 802.1x authentication.

```
device(config)# aaa authentication dot1x default radius
```

The following example enables MAC authentication.

```
device(config)# aaa authentication dot1x default radius
```

# aaa authentication enable

Configures the AAA authentication method for securing access to the Privileged EXEC level and global configuration levels of the CLI.

## Syntax

**aaa authentication enable default** *method-list* [ *method-list* ... ]

**no aaa authentication enable default** *method-list* [ *method-list* ... ]

**aaa authentication enable implicit-user**

**no aaa authentication enable implicit-user**

## Command Default

The AAA authentication method list is not configured.

By default, the device prompts for a username and password.

## Parameters

**default**

Configures the default authentication method list.

*method-list*

Configures the following authentication methods.

**enable**

Authenticate using the password you configured for the Super User privilege level. This password is configured using the **enable super-user-password** command.

**line**

Authenticate using the password you configured for Telnet access. The Telnet password is configured using the **enable telnet password** command.

**local**

Authenticate using a local username and password you configured on the device. Local usernames and passwords are configured using the **username** command.

**none**

Does not use any authentication method. The device automatically permits access.

**radius**

Authenticate using the database on a RADIUS server. You also must identify the server to the device using the **radius-server** command.

**tacacs**

Authenticate using the database on a TACACS server. You also must identify the server to the device using the **tacacs-server** command.

**tacacs+**

Authenticate using the database on a TACACS+ server. You also must identify the server to the device using the **tacacs-server** command.

**implicit-user**

Configures the device to prompt only for a password when a user attempts to gain Super User access to the Privileged EXEC and global configuration levels of the CLI.

# Modes

Global configuration mode

# Usage Guidelines

You can specify a primary authentication method and up to six backup authentication methods. If the configured primary authentication fails due to an error, the device tries the backup authentication methods in the order they appear in the list.

If enable authentication is configured on the device, when a user attempts to gain Super User access to the Privileged EXEC and global configuration levels of the CLI, by default the device prompts for a username and password. You can configure the device to prompt only for a password. The device uses the username entered at login, if one is available. If no username was entered at login, the device prompts for both username and password.

The **no** form of the command removes authentication method.

# Examples

The following example shows how to configure TACACS/TACACS+ as the primary authentication method for securing access to the Privileged EXEC and global configuration levels of the CLI. In this example, TACACS/TACACS+ is configured to be the primary authentication method for securing access. If TACACS/TACACS+ authentication fails due to an error with the server, local authentication is used instead. If local authentication fails, no authentication is used; the device automatically permits access.

```
device(config)# aaa authentication enable default tacacs local none
```

The following example shows how to configure RADIUS as the primary authentication method and other backup authentication methods.

```
device(config)# aaa authentication enable default radius tacacs tacacs+ enable local line none
```

The following example shows how to configure the device to prompt only for a password when a user attempts to gain Super User access to the Privileged EXEC and global configuration levels of the CLI.

```
device(config)# aaa authentication enable implicit-user
```

# aaa authentication login

Configures the AAA authentication method for securing access to Telnet or SSH access to the CLI.

## Syntax

**aaa authentication login default** *method-list* [ *method-list* ... ]

**no aaa authentication login default** *method-list* [ *method-list* ... ]

**aaa authentication login privilege-mode**

**no aaa authentication login privilege-mode**

## Command Default

The AAA authentication method list is not configured.

By default, a user enters the User EXEC mode after a successful login through Telnet or SSH.

## Parameters

**default**

Configures the default authentication method list.

*method-list*

Configures the following authentication methods.

**enable**

Authenticate using the password you configured for the Super User privilege level. This password is configured using the **enable super-user-password** command.

**line**

Authenticate using the password you configured for Telnet access. The Telnet password is configured using the **enable telnet password** command.

**local**

Authenticate using a local username and password you configured on the device. Local usernames and passwords are configured using the **username** command.

**none**

Does not use any authentication method. The device automatically permits access.

**radius**

Authenticate using the database on a RADIUS server. You also must identify the server to the device using the **radius-server** command.

**tacacs**

Authenticate using the database on a TACACS server. You also must identify the server to the device using the **tacacs-server** command.

**tacacs+**

Authenticate using the database on a TACACS+ server. You also must identify the server to the device using the **tacacs-server** command.

**privilege-mode**

Configures the device to enter the privileged EXEC mode after a successful login through Telnet or SSH.

# Modes

Global configuration mode

# Usage Guidelines

You can specify a primary authentication method and up to six backup authentication methods. If the configured primary authentication fails due to an error, the device tries the backup authentication methods in the order they appear in the list.

By default, a user enters User EXEC mode after a successful login through Telnet or SSH. Optionally, you can configure the device so that a user enters Privileged EXEC mode after a Telnet or SSH login. The user privilege level is based on the privilege level granted during login.

The **no** form of the command removes the authentication method.

# Examples

The following example shows how to configure RADIUS as the primary authentication method for securing Telnet access to the CLI. If RADIUS authentication fails due to an error with the server, local authentication is used instead.

```
device(config)# aaa authentication login default radius local
```

The following example shows how to configure RADIUS as the primary authentication method and other backup authentication methods.

```
device(config)# aaa authentication login default radius tacacs tacacs+ enable local line none
```

The following example shows how to configure the device so that a user enters Privileged EXEC mode after a Telnet or SSH login.

```
device(config)# aaa authentication login privilege-mode
```

# aaa authentication snmp-server

Configures the AAA authentication method for SNMP server access.

## Syntax

**aaa authentication snmp-server default** *method-list* [ *method-list* ... ]

**no aaa authentication snmp-server default** *method-list* [ *method-list* ... ]

## Command Default

The AAA authentication method list is not configured.

## Parameters

**default**

Configures the default authentication method list.

*method-list*

Configures the following authentication methods.

**enable**

Authenticate using the password you configured for the Super User privilege level. This password is configured using the **enable super-user-password** command.

**line**

Authenticate using the password you configured for Telnet access. The Telnet password is configured using the **enable telnet password** command

**local**

Authenticate using a local username and password you configured on the device. Local usernames and passwords are configured using the **username** command.

**none**

Does not use any authentication method. The device automatically permits access.

**radius**

Authenticate using the database on a RADIUS server. You also must identify the server to the device using the **radius-server** command.

**tacacs**

Authenticate using the database on a TACACS server. You also must identify the server to the device using the **tacacs-server** command.

**tacacs+**

Authenticate using the database on a TACACS+ server. You also must identify the server to the device using the **tacacs-server** command.

## Modes

Global configuration mode

# Usage Guidelines

You can specify a primary authentication method and up to six backup authentication methods. If the configured primary authentication fails due to an error, the device tries the backup authentication methods in the order they appear in the list.

When this command is enabled, community string validation is not performed for incoming SNMP v1and v2c packets. This command takes effect as long as the first varbind for SNMP packets is set to one of the following:

- snAgGblPassword=" username password " (for AAA method local)
- snAgGblPassword=" password " (for AAA method line, enable)

**NOTE**
Certain SNMP objects need additional validation. These objects include but are not limited to: snAgReload, snAgWriteNVRAM, snAgConfigFromNVRAM, snAgImgLoad, snAgCfgLoad, and snAgGblTelnetPassword.

If AAA is set up to check both the username and password, the string contains the username, followed by a space and then the password. If AAA is set up to authenticate with the current Enable or Line password, the string contains the password only. The configuration can be overridden by the **no snmp-server pw-check** command, which disables password checking for SNMP SET requests.

The **no** form of the command removes the authentication method.

# Examples

The following example shows how to configure incoming SNMP SET operations to be authenticated using the locally configured usernames and passwords.

```
device(config)# aaa authentication snmp-server default local
```

# aaa authentication web-server

Configures the AAA authentication method to access the device through the Web Management Interface.

## Syntax

**aaa authentication web-server default** *method-list* [ *method-list* ... ]

**no aaa authentication web-server default** *method-list* [ *method-list* ... ]

## Command Default

The AAA authentication is not configured.

## Parameters

**default**

Configures the default authentication method list.

*method-list*

Configures the following authentication methods.

**enable**

Authenticate using the password you configured for the Super User privilege level. This password is configured using the **enable super-user-password** command.

**line**

Authenticate using the password you configured for Telnet access. The Telnet password is configured using the **enable telnet password** command.

**local**

Authenticate using a local username and password you configured on the device. Local usernames and passwords are configured using the **username** command.

**none**

Does not use any authentication method. The device automatically permits access.

**radius**

Authenticate using the database on a RADIUS server. You also must identify the server to the device using the **radius-server** command.

**tacacs**

Authenticate using the database on a TACACS server. You also must identify the server to the device using the **tacacs-server** command.

**tacacs+**

Authenticate using the database on a TACACS+ server. You also must identify the server to the device using the **tacacs-server** command.

## Modes

Global configuration mode

# Usage Guidelines

You can specify a primary authentication method and up to six backup authentication methods. If the configured primary authentication fails due to an error, the device tries the backup authentication methods in the order they appear in the list.

The **no** form of the command removes authentication method.

# Examples

The following example shows how to configure the device to use the local user accounts to authenticate access to the device through the Web Management Interface. If the device does not have a user account that matches the username and password entered by the user, the user is not granted access.

```
device(config)# aaa authentication web-server default local
```

# aaa authorization coa enable

Enables RADIUS Change of Authorization (CoA).

## Syntax

**aaa authorization coa enable**

**no aaa authorization coa enable**

## Command Default

RADIUS CoA is not enabled.

## Parameters

None

## Modes

Global configuration mode

## Usage Guidelines

Use this command to enable RADIUS CoA authorization. The no form of the command disables the CoA functionality. A change of authorization request packet can be sent by the Dynamic Authorization Client (DAC) to change the session authorizations on the Network Access Server (NAS). This is used to change the filters, such as Layer 3 ACLs.

Before RFC 5176 when a user or device was authenticated on the RADIUS server, the session could only be ended if the user or device logs out. RFC 5176 addresses this issue by adding two more packet types to the current RADIUS standard: Disconnect Message and Change of Authorization. The Dynamic Authorization Client (DAC) server makes the requests to either delete the previously established sessions or replace the previous configuration or policies. Currently, these new extensions can be used to dynamically terminate or authorize sessions that are authenticated through multi-device-port-authentication or dot1x authentication.

## Examples

The following example enables RADIUS CoA.

```
device(config)# aaa authorization coa enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# aaa authorization coa ignore

Discards the specified RADIUS Change of Authorization (CoA) messages.

## Syntax

**aaa authorization coa ignore** { **disable-port** | **dm-request** | **flip-port** | **modify-acl** | **reauth-host** }

**no aaa authorization coa ignore** { **disable-port** | **dm-request** | **flip-port** | **modify-acl** | **reauth-host** }

## Command Default

The default state is maintained, and the packets are not discarded. All options are enabled for CoA processing.

## Parameters

**disable-port**
Disables the port.

**dm-request**
Disconnects the message request.

**flip-port**
Toggles the port.

**modify-acl**
Modifies the access control list.

**reauth-host**
Reauthenticates the host.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to discard the specified RADIUS messages. A CoA request packet can be sent by the Dynamic Authorization Client (DAC) to change the session authorizations on the Network Access Server (NAS). This is used to change the filters, such as Layer 3 ACLs.

Before RFC 5176, when a user or device was authenticated on the RADIUS server, the session could be ended only if the user or device logs out. RFC 5176 addresses this issue by adding two more packet types to the current RADIUS standard: Disconnect Message and Change of Authorization. The Dynamic Authorization Client (DAC) server makes the requests to either delete the previously established sessions or replace the previous configuration or policies. Currently, these new extensions can be used to dynamically terminate or authorize sessions that are authenticated through MAC authentication or 802.1X authentication.

The **no** form of the command honors the dm-request message.

# Examples

The following example ignores the disconnect message request.

```
device(config)# aaa authorization coa ignore dm-request
```

The following example ignores the host reauthentication message request.

```
device(config)# aaa authorization coa ignore reauth-host
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | This command was updated with **disable-port**, **flip-port**, and **reauth-host** options. |

# aaa authorization commands

Configures the AAA authorization configuration parameters for EXEC commands.

## Syntax

**aaa authorization commands** *privilege-level* **default radius** [ **tacacs+** ] [ **none** ]

**no aaa authorization commands** *privilege-level* **default radius** [ **tacacs+** ] [ **none** ]

**aaa authorization commands** *privilege-level* **default tacacs+** [ **radius** ] [ **none** ]

**no aaa authorization commands** *privilege-level* **default tacacs+** [ **radius** ] [ **none** ]

**aaa authorization commands** *privilege-level* **default none**

**no aaa authorization commands** *privilege-level* **default none**

## Command Default

AAA authorization is not enabled.

## Parameters

*privilege-level*

Configures the device to perform AAA authorization for the commands available at the specified privilege level. Valid values are 0 (Super User level - all commands), 4 (Port Configuration level - port-config and read-only commands), and 5 (Read Only level - read-only commands).

**default**

Configures the default named list.

**radius**

Configures RADIUS authorization.

**tacacs+**

Configures TACACS+ authorization.

**none**

Disables accounting.

## Modes

Global configuration mode

## Usage Guidelines

You can configure RADIUS, TACACS+, and None as authorization methods. If the configured primary authorization fails due to an error, the device tries the backup authorization methods in the order they are configured.

When TACACS+ command authorization is enabled, the ICX device consults a TACACS+ server to get authorization for commands entered by the user.

When RADIUS command authorization is enabled, the ICX device consults the list of commands supplied by the RADIUS server during authentication to determine whether a user can issue a command that was entered.

**NOTE**
TACACS+ and RADIUS command authorization can be performed only for commands entered from Telnet or SSH sessions, or from the console. No authorization is performed for commands entered at the Web Management Interface.

TACACS+ command authorization is not performed for the following commands:

- At all levels: **exit**, **logout**, **end**, and **quit**.
- At the Privileged EXEC level: **enable** or **enable** *text*, where *text* is the password configured for the Super User privilege level.

Because RADIUS command authorization relies on the command list supplied by the RADIUS server during authentication, you cannot perform RADIUS authorization without RADIUS authentication.

The **no** form of the command disables authorization.

# Examples

The following example shows how to configure RADIUS command authorization for the commands available at the Super User privilege level (that is, all commands on the device).

```
device(config)# aaa authorization commands 0 default radius
```

The following example shows how to configure TACACS+ command authorization for the commands available at the Super User privilege level (that is, all commands on the device).

```
device(config)# aaa authorization commands 0 default tacacs+
```

# aaa authorization exec

Determines the user privilege level when users are authenticated.

## Syntax

**aaa authorization exec default radius** [ **tacacs+** ] [ **none** ]

**no aaa authorization exec default radius** [ **tacacs+** ] [ **none** ]

**aaa authorization exec default tacacs+** [ **radius** ] [ **none** ]

**no aaa authorization exec default tacacs+** [ **radius** ] [ **none** ]

**aaa authorization exec default none**

**no aaa authorization exec default none**

## Command Default

AAA authorization is not configured.

## Parameters

**default**
Configures the default named list.

**radius**
Configures RADIUS authorization.

**tacacs+**
Configures TACACS+ authorization.

**none**
Disables accounting.

## Modes

Global configuration mode

## Usage Guidelines

You can configure RADIUS, TACACS+, and None as authorization methods. If the configured primary authorization fails due to an error, the device tries the backup authorization methods in the order they are configured.

When TACACS+ EXEC authorization is performed, the ICX device consults a TACACS+ server to determine the privilege level of the authenticated user. If the **aaa authorization exec default tacacs+** command exists in the configuration, following successful authentication, the device assigns the user the privilege level specified by the foundry-privilege-level received from the TACACS+ server. If the **aaa authorization exec default tacacs+** command does not exist in the configuration, then the value in the foundry-privilege-level attribute is ignored, and the user is granted Super User access. Also note that in order for the **aaa authorization exec default tacacs+** command to work, either the **aaa**

**authentication enable default tacacs+** command, or the **aaa authentication login privilege-mode** command must also exist in the configuration.

When RADIUS EXEC authorization is performed, the ICX device consults a RADIUS server to determine the privilege level of the authenticated user. If the **aaa authorization exec default radius** command exists in the configuration, following successful authentication, the device assigns the user the privilege level specified by the foundry-privilege-level attribute received from the RADIUS server. If the **aaa authorization exec default radius** command does not exist in the configuration, then the value in the foundry-privilege-level attribute is ignored, and the user is granted Super User access. Also note that in order for the **aaa authorization exec default radius** command to work, either the **aaa authentication enable default radius** command, or the **aaa authentication login privilege-mode** command must also exist in the configuration.

The **no** form of the command disables authorization.

# Examples

The following example shows how to configure TACACS+ EXEC authorization.

```
device(config)# aaa authorization exec default tacacs+
```

The following example shows how to configure RADIUS EXEC authorization.

```
device(config)# aaa authorization exec default radius
```

# accept-lifetime

Configures the time period during which the key on a keychain becomes active and is received as valid.

## Syntax

**accept-lifetime** [ **local** | **start** { *start-date start-time* **end** { *duration* | **infinite** | *end-date end-time* } } ]

**no accept-lifetime**

## Command Default

The lifetime of accept keys is not configured by default.

## Parameters

**local**
Specifies that the time zone used will be the time zone configured in the system.

**start**
Configures the point of time from which the key is received as valid.

*start-date*
Configures the start date in the *dd-mm-yy* format.

*start-time*
Configures the start time in the *hh:mm:ss* format.

**end**
Configures the point of time at which the accept key expires.

*duration*
Configures the duration in seconds before the accept key expires. The value ranges from 1 through 2147483646 seconds.

**infinite**
Configures the accept key to never expire.

*end-date*
Configures the end date in the *dd-mm-yy* format.

*end-time*
Configures the end time in the *hh:mm:ss* format.

## Modes

Key ID configuration mode

## Usage Guidelines

All participating routers must have Network Time Protocol (NTP) enabled before setting the lifetime on the keys.

If the tolerance value is configured, the start time of the accept key to become active is advanced (start time minus tolerance) and the end time is moved further ahead (end time plus tolerance) before the key expires, unless the end-time is set to be infinite.

A key is considered valid even when it is in the tolerance period.

A key can be selectively active for the send lifetime and not the accept lifetime.

The key must be configured with a minimum time of ten seconds.

The **no** form of the command negates the entire accept lifetime and not merely individual options of the duration.

# Examples

The following example configures the time period during which the key on a keychain becomes active and is received as a valid key.

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)# key-id 10
device(config-keychain-xprotocol-key-10)# accept-lifetime start 10-10-17 10:10:10 end 10000
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# accept-mode

Enables a backup VRRP device to respond to ping, traceroute, and Telnet packets if the backup device becomes the master VRRP device.

## Syntax

**accept-mode**

**no accept-mode**

## Command Default

A VRRP nonowner master router does not respond to any packet that is destined for the virtual IPv4 or IPv6 address.

## Modes

VRID interface configuration mode

## Usage Guidelines

The **no** form of this command causes the nonowner master router to not respond to any packet that is destined for the virtual IPv4 or IPv6 address of the VRRP session.

A VRRP nonowner master router does not respond to any packet that is destined for the virtual IPv4 or IPv6 address. This prevents troubleshooting network connections to this router using ping, traceroute, or Telnet. To resolve this, you can use this command to enable the router to respond to ping, traceroute, and Telnet packets destined for the virtual IPv4 or IPv6 address of a VRRP cluster. The router drops all other packets destined for the virtual IPv4 or IPv6 address of the VRRP session.

**NOTE**
The **accept-mode** command enables the device to respond to ping, traceroute, and Telnet packets, but the device does not respond to SSH packets. When the device acting as the master router is not the IP address owner (the router with the interface whose actual IP address is used as the virtual router's IP address), the master router accepts only the ARP packets sent to the virtual IP address. When accept mode is configured, the master router responds to ping, TELNET, and traceroute packets sent to the virtual IP address even when the master router is not the IP address owner.

## Examples

The following example shows the configuration of accept mode on an IPv6 VRRP backup router.

```
device# configure terminal
device(config)# interface ve 3
device(config-vif-3)# ipv6 vrrp vrid 2
device(config-vif-3-vrid-2)# backup
device(config-vif-3-vrid-2)# advertise backup
device(config-vif-3-vrid-2)# ipv6-address 2001:DB8::1
device(config-vif-3-vrid-2)# accept-mode
device(config-vif-3-vrid-2)# activate
```

# History

| Release version | Command history |
|---|---|
| 8.0.01 | This command was introduced. |
| 8.0.30b | This command was modified to explain that the **accept-mode** command does not enable a response to SSH packets. The usage guidelines were also updated. |

# access-control vlan

Enables the VLAN containment for Network Time Protocol (NTP).

## Syntax

**access-control vlan** *vlan-id*

**no access-control vlan** *vlan-id*

## Command Default

VLAN containment for NTP is not enabled.

## Parameters

*vlan-id*

Specifies the VLAN ID number.

## Modes

NTP configuration mode

## Usage Guidelines

The management interface is not part of any VLAN. When configuring the VLAN containment for NTP, the management interface is not used to send or receive the NTP packets.

When VLAN is configured,

- NTP time servers should be reachable through the interfaces that belong to the configured VLAN. Otherwise, NTP packets are not transmitted. NTP packets are not trasmitted in the case of both the unicast and the broadcast server/client if the servers are not reachable through the interfaces that belong to the configured VLAN.
- NTP broadcast packets are sent only on the interface that belongs to the configured VLAN.
- The received unicast or broadcast NTP packets are dropped if the interface on which the packets have been received does not belong to the configured VLAN.

The **no** form of the command removes the specified VLAN containment for NTP.

## Examples

The following example enables VLAN containment for NTP.

```
device(config)# ntp
device(config-ntp)# access-control vlan 100
```

# accounting

Enables RADIUS accounting for Web Authentication.

## Syntax

**accounting**

**no accounting**

## Command Default

RADIUS accounting for Web Authentication is not enabled.

## Modes

Web Authentication configuration mode

## Usage Guidelines

When Web Authentication is enabled, you can enable RADIUS accounting to record login (start) and logout (stop) events per host. The information is sent to a RADIUS server.

The **no** form of the command disables RADIUS accounting for Web Authentication.

## Examples

The following example enables RADIUS accounting for Web Authentication.

```
device(config)# vlan 10
device(config-vlan-10# webauth
device(config-vlan-10-webauth)# accounting
```

# acl-logging

Enables IPv4 ACL logging for ACL permit and deny ACL rules that contain the **log** keyword.

## Syntax

**acl-logging**

**no acl-logging**

## Command Default

IPv4 ACL logging is disabled.

## Modes

Interface configuration modes

## Usage Guidelines

This command enables ACL logging only on physical and virtual interfaces to which it is applied.

ACL logging is supported for both deny rules and permit rules.

ACL logging is supported for both ingress and egress ACLs.

ACL logging is a CPU-intensive feature intended for debugging purposes. Ruckus recommends that you disable ACL logging after the debug session is over.

ACL logging is not supported for dynamic ACLs with MAC authentication or 802.1X enabled.

The **acl-logging** command applies to IPv4 ACLs only. For IPv6 ACLs, refer to the **logging-enable** command.

The **no** form of the command disables IPv4 ACL logging on the current interface.

## Examples

The following example enables ACL logging for the 1/1/4 interface. Because the ACL bound to that interface includes rules with the **log** keyword, packets that match any such rules are logged.

```
device# configure terminal
device(config)# ip access-list standard 1
device(config-std-nacl)# deny host 10.157.22.26 log
device(config-std-nacl)# deny 10.157.29.12 log
device(config-std-nacl)# deny host IPHost1 log
device(config-std-nacl)# permit any
device(config-std-nacl)# interface ethernet 1/1/4
device(config-if-e1000-1/1/4)# acl-logging
device(config-if-e1000-1/1/4)# ip access-group 1 in
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to support permit rules (in addition to deny rules) and egress ACLs (in addition to ingress ACLs). |

# acl-mirror-port

Configures ACL-based inbound mirroring.

## Syntax

**acl-mirror-port ethernet** *unit/slot/port*

**no acl-mirror-port ethernet** *unit/slot/port*

## Parameters

**ethernet** *unit/slot/port*

Specifies the mirror port to which the monitored port traffic is copied.

## Modes

Interface configuration mode

## Usage Guidelines

Use this command to set the destination port on which the traffic must be mirrored. The destination port must be the same for all ports in a port region. All traffic mirrored from any single port in a port region is mirrored to the same destination mirror port as traffic mirrored from any other port in the same port region. When a destination port is configured for any port within a port region, traffic from any ACL with a mirroring clause assigned to any port in that port region is mirrored to that destination port. This will occur even if a destination port is not explicitly configured for the port with the ACL configured.

To configure ACL-based mirroring for ACLs bound to virtual interfaces, use the **acl-mirror-port** command on a physical port that is a member of the same VLAN as the virtual interface. You can apply ACL-based mirroring on an entire VE, and enable mirroring in only one port region; traffic that is in the same VE but on a port in a different port region will not be mirrored. If a port is in both mirrored and non-mirrored VLANs, only traffic on the port from the mirrored VLAN is mirrored.

**NOTE**
If a destination mirror port is not configured for any ports within the port region where the port-mirroring ACL is configured, the ACL does not mirror the traffic but the ACL is applied to traffic on the port.

The **no** form of the command removes the ACL mirror port.

## Examples

The following example shows the ACL mirroring traffic from port 1/1/1 is mirrored to port 1/1/3.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# acl-mirror-port ethernet 1/1/3
```

The following example shows that ports from a port region must be mirrored to the same destination mirror port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# acl-mirror-port ethernet 1/2/3
device(config)# interface ethernet 1/1/2
device(config-if-e10000-1/1/2)# acl-mirror-port ethernet 1/2/3
```

The following example shows ACL mirroring when the destination port within a port region is configured.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# ip access-group 101 in
device(config)# interface ethernet 1/1/3
device(config-if-e10000-1/1/3)# acl-mirror-port ethernet 1/4/3
```

The following example shows how to specify the destination mirror port for LAG ports.

```
device(config)# lag blue static id 1
device(config-lag-blue)# ports ethernet 1/1/1 to 1/1/14
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# acl-mirror-port ethernet 1/1/8
```

The following example shows how to configure ACL-based mirroring for ACLs bound to virtual interfaces.

```
device(config)# vlan 10
device(config-vlan-10)# tagged ethernet 1/4/1 to 1/4/2
device(config-vlan-10)# tagged ethernet 1/5/3
device(config-vlan-10)# router-interface ve 10
device(config-vlan-10)# exit
device(config)# interface ethernet 1/4/1
device(config-if-e10000-1/4/1)# acl-mirror-port ethernet 1/5/1
device(config-if-e10000-1/4/1)# exit
device(config)# interface ve 10
device(config-vif-10)# ip address 10.10.10.254/24
device(config-vif-10)# ip access-group 102 in
device(config)# access-list 102 permit ip any any mirror
```

The following example shows the ACL-based mirroring for ports in both mirrored and non-mirrored VLANs.

```
device(config)# vlan 10
device(config-vlan-10)# tagged ethernet 1/4/1 to 1/4/2
device(config-vlan-10)# tagged ethernet 1/5/3
device(config-vlan-10)# router-interface ve 10
device(config-vlan-10)# exit
device(config)# vlan 20
device(config-vlan-20)# tagged ethernet 1/4/1 to 1/4/2
device(config-vlan-20)# exit
device(config)# interface ethernet 1/4/1
device(config-if-e10000-1/4/1)# acl-mirror-port ethernet 1/5/1
device(config-if-e10000-1/4/1)# exit
device(config)# interface ve 10
device(config-vif-10)# ip address 10.10.10.254/24
device(config-vif-10)# ip access-group 102 in
device(config-vif-10)# exit
device(config)# access-list 102 permit ip any any mirror
```

# activate (VRRP)

Activates the configured Virtual Router Redundancy Protocol (VRRP) virtual routing instance.

## Syntax

**activate**

**no activate**

## Command Default

A VRRP virtual routing instance is not activated.

## Modes

VRID interface configuration mode

## Usage Guidelines

Before issuing this command, complete the configuration of the VRRP virtual router. The interface assigned to the Virtual Routing ID (VRID) does not provide backup service for the virtual IP address until you activate the VRRP configuration.

The **no** form of this command disables the VRRP VRID.

## Examples

The following example configures and activates VRRP VRID 1.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.5.1
device(config-if-e1000-1/1/6-vrid-1)# activate
VRRP router 1 for this interface is activating
```

# activate (VSRP)

Activates the Virtual Switch Redundancy Protocol (VSRP) Virtual Router ID (VRID) for a port-based VLAN.

## Syntax

**activate**

**no activate**

## Command Default

The VRID is not activated by default.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The device must be set as a backup. Because VSRP does not have an owner, all VSRP devices are backups. The active device for a VRID is elected based on the VRID priority, which is configurable.

The **no** form of the command deactivates the VSRP VRID on the VLAN.

## Examples

The following example shows how to activate the VSRP on a VLAN.

```
device(config)# vlan 200
device(config-vlan-200)# tag ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup
device(config-vlan-200-vrid-1)# activate
```

# auto-enroll (PKI)

Sends enrollment message to the CA and local auto-enroll certificates.

## Syntax

**auto-enroll** { *renewal_percentage* | **regenerate** }

**no auto-enroll** { *renewal_percentage* | **regenerate** }

## Command Default

Auto-enrollment is not enabled.

## Parameters

*renewal_percentage*

Sets the renewal percentage. The valid range is 10 through 90 per cent. The default is 80 per cent.

**regenerate**

Generates a new key pair for the certificates.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command disables auto-enrollment.

## Examples

The following example enables auto-enrollment.

```
device# configure terminal
device(config)# pki trustpoint trust1
device(config-pki-trustpoint-trust1)# auto-enroll
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# add mac

Permanently authenticates certain hosts.

## Syntax

**add mac** *mac-address* [ **ethernet** *unit/slot/port* ] [ **duration** *time* ]

**no add mac** *mac-address* [ **ethernet** *unit/slot/port* ] [ **duration** *time* ]

## Command Default

Permanent authentication is not enabled.

## Parameters

*mac-address*

Specifies the MAC address of the host.

**ethernet** *unit/slot/port*

Specifies the Ethernet interface.

**duration** *time*

Specifies how long the MAC address remains authenticated. Valid values are from 0 through 128,000 seconds. The default is the time configured using the **reauth-time** command. If 0 is configured, then Web Authentication for the MAC address will not expire.

## Modes

Web Authentication configuration mode

## Usage Guidelines

Certain hosts, such as a DHCP server, gateway, or printers, may need to be permanently authenticated. Typically, these hosts are managed by the network administrator and are considered to be authorized hosts. Also, some of these hosts (such as printers) may not have a browser and will not be able to perform the Web Authentication.

**NOTE**

If a MAC address is statically configured, the MAC address will not be allowed to be dynamically configured on any port.

The **no** form of the command, without any parameters, removes all hosts and sets the duration a MAC address remains authenticated to its default.

# Examples

The following example configures the host with MAC address 0000.00eb.2d14 to be permanently authenticated.

```
device(config)# vlan 10
device(config-vlan-10# webauth
device(config-vlan-10-webauth)# add mac 0000.00eb.2d14 duration 0
```

The following example specifies the MAC address to be added by the specified port that is a member of the VLAN.

```
device(config)# vlan 10
device(config-vlan-10# webauth
device(config-vlan-10-webauth)# add mac 0000.00eb.2d14 ethernet 1/1/1 duration 0
```

# add-vlan

Adds individual VLANs or a range of VLANs.

## Syntax

**add-vlan** *vlan-id* [ **to** *vlan-id* ]

## Command Default

VLANs are added when creating a VLAN group.

## Parameters

*vlan-id*

Specifies the VLAN ID to add.

**to** *vlan-id*

Specifies the range of VLANs to add.

## Modes

VLAN group configuration mode

## Usage Guidelines

Use the **vlan-group** command to add up to 256 VLANs. To add more than 256 VLANs, use the **add-vlan** command.

**NOTE**
The device memory must be configured to contain at least the number of VLANs you specify for the higher end of the range. For example, if you specify 2048 as the VLAN ID at the high end of the range, you first must increase the memory allocation for VLANs to 2048 or higher. Additionally, on Layer 3 switches, if you allocate additional memory for VLANs, you also need to allocate the same amount of memory for virtual routing interfaces before you configure the VLAN groups. This is true regardless of whether you use the virtual routing interface groups. The memory allocation is required because the VLAN groups and virtual routing interface groups have a one-to-one mapping.

## Examples

The following example shows how to add VLANs.

```
device(config)# vlan-group 1 vlan 2 to 1000
device(config-vlan-group-1)# add-vlan 1001 to 1002
```

# address-family

Enables IPv4 or IPv6 address-family configuration mode.

## Syntax

**address-family** { **ipv4** | **ipv6** } [ **max-route** *num* ]

**no address-family** { **ipv4** | **ipv6** } [ **max-route** *num* ]

## Command Default

An address family is not configured.

## Parameters

**ipv4**

Specifies an IPv4 address family.

**ipv6**

Specifies an IPv6 address family.

**max-route** *num*

Configures the maximum number routes in a VRF. The valid range is from 128 through 15168. The default is 1024.

## Modes

VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to remove IPv4 or IPv6 address family configurations from the device.

## Examples

The following example enables IPv4 address-family configuration mode:

```
device(config)# vrf red
device(config-vrf-red)# address-family ipv4
device(config-vrf-red-ipv4)#
```

# address-family unicast (BGP)

Enables the IPv4 or IPv6 address family configuration mode to configure a variety of BGP4 unicast routing options.

## Syntax

**address-family ipv4 unicast vrf** *vrf-name*

**address-family ipv6 unicast** [ **vrf** *vrf-name* ]

**no address-family ipv4 unicast vrf** *vrf-name*

**no address-family ipv6 unicast** [ **vrf** *vrf-name* ]

## Parameters

**ipv4**

Specifies an IPv4 address family.

**ipv6**

Specifies an IPv6 address family.

**vrf** *vrf-name*

Specifies the name of the VRF instance to associate with subsequent address-family configuration mode commands.

## Modes

BGP configuration mode

## Usage Guidelines

Use the **no** form of this command to remove IPv4 or IPv6 address family configurations from the device.

## Examples

This example enables BGP IPv6 address family configuration mode.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)#
```

This example creates a BGP4 unicast instance for VRF green.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf green
device(config-bgp-ipv4u-vrf)#
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Multi-VRF support was added for IPv6 BGP. |

# advertise backup

Advertises a Virtual Router Redundancy Protocol (VRRP) backup router to a VRRP master router.

## Syntax

**advertise backup**

**no advertise backup**

## Command Default

A VRRP backup router does not advertise itself to a VRRP master router.

## Modes

VRID interface configuration mode

## Usage Guidelines

Hello messages are used to advertise a backup router to a master router. To configure the interval at which the messages are sent, use the **backup-hello-interval** command.

The **advertise backup** command is configured only on VRRP backup routers and is supported by VRRP and VRRP-E.

The **no** form of the command disables the advertisement of a VRRP backup router to a VRRP master router.

## Examples

The following example enables advertisements from the VRRP backup router and configures the hello message interval to 10 seconds.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# advertise backup
device(config-if-e1000-1/1/6-vrid-1)# backup-hello-interval 10
```

# advertise backup (VSRP)

Enables a backup to send Hello messages to the master.

## Syntax

**advertise backup**

**no advertise backup**

## Command Default

By default, backups do not send Hello messages to advertise themselves to the master.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

When a backup is enabled to send Hello messages, the backup sends a Hello message to the master every 60 seconds by default. You can change the interval to be up to 3600 seconds using the **backup-hello-interval** command.

The **no** form of the command disables the backup from sending the Hello messages.

## Examples

The following example enables a backup to send Hello messages to the master.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup
device(config-vlan-200-vrid-1)# activate
device(config-vlan-200-vrid-1)# advertise backup
```

# age

Configures the device to age out secure MAC addresses after a specified amount of time.

## Syntax

**age** { **global-mac** | *time* [ **absolute** ] }

**no age** { **global-mac** | *time* [ **absolute** ] }

## Command Default

By default, learned MAC addresses stay secure indefinitely.

## Parameters

**global-mac**

Configures hardware-based aging of all secure MAC addresses.

*time*

Configures the age timer. Valid values range is from 0 through 1440 minutes. If 0 is specified, the MAC addresses stay secure indefintely.

**absolute**

Configures all secure MAC addresses to age out immediately once the specified time expires.

## Modes

Port security configuration mode

Port security interface configuration mode

## Usage Guidelines

If the **absolute** keyword is not specified, secure MAC addresses are aged out only when the configured hardware MAC address age time expires.

**NOTE**
Even though you can set the age time to specific ports independent of the device-level setting, the age timer will take the greater of the two values. If you set the age timer to 3 minutes for the port, and 10 minutes for the device, the port MAC address aging occurs in 10 minutes (the device-level setting), which is greater than the port setting that you have configured.

On the ICX 7750, the port security age can only be set to the global hardware age. The absolute age and no aging of secure MACs are configured as static in hardware.

The **no** form of the command configures to never age out secure MAC addresses.

# Examples

The following example sets the port security age timer to 10 minutes on all interfaces.

```
device(config)# port security
device(config-port-security)# age 10
```

The following example ages out secure MAC addresses immediately after one minute.

```
device(config)# port security
device(config-port-security)# age 1 absolute
```

The following example sets the port security age timer to 10 minutes on a specific interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)#port security
device(config-port-security-e1000-1/1/1)# age 10
```

# aggregate-address (BGP)

Configures the device to aggregate routes from a range of networks into a single network prefix.

## Syntax

**aggregate-address** { *ip-addr ip-mask* | *ipv6-addr ipv6-mask* } [ **advertise-map** *map-name* ] [ **as-set** ] [ **attribute-map** *map-name* ] [ **summary-only** ] [ **suppress-map** *map-name* ]

**no aggregate-address** { *ip-addr ip-mask* | *ipv6-addr ipv6-mask* } [ **advertise-map** *map-name* ] [ **as-set** ] [ **attribute-map** *map-name* ] [ **summary-only** ] [ **suppress-map** *map-name* ]

## Command Default

The address aggregation feature is disabled. By default, the device advertises individual routes for all networks.

## Parameters

*ip-addr*
IPv4 address.

*ip-mask*
IPv4 mask.

*ipv6-addr*
IPv6 address.

*ipv6-mask*
IPv6 mask.

**advertise-map**
Causes the device to advertise the more-specific routes in the specified route map.

*map-name*
Specifies a route map to be consulted.

**as-set**
Causes the device to aggregate AS-path information for all routes in the aggregate routes from a range of networks into a single network prefix.

**attribute-map**
Causes the device to set attributes for the aggregate routes according to the specified route map.

**summary-only**
Prevents the device from advertising more-specific routes contained within the aggregate route.

**suppress-map**
Prevents the more-specific routes contained in the specified route map from being advertised.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

# Usage Guidelines

Use the **no** form of this command to restore the defaults. When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

# Examples

This example aggregates routes from a range of networks into a single network prefix and prevents the device from advertising more-specific routes.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# aggregate-address 10.11.12.0 summary-only
```

This example aggregates routes from a range of networks into a single network prefix under the IPv6 address family and advertises the paths for this route as AS_SET.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# aggregate-address 2001:DB8:12D:1300::/64 as-set
```

This example aggregates routes from a range of networks into a single network prefix for BGP VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# aggregate-address 5.0.0.0/8
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# aggregated-vlan

Enables support for larger Ethernet frames.

## Syntax

**aggregated-vlan**

**no aggregated-vlan**

## Command Default

Support for larger Ethernet frames is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

This command provides support for Ethernet frames up to 1536 bytes.

The **no** form of the command disables support for larger Ethernet frames.

## Examples

The following example provides support for larger Ethernet frames.

```
device(config)# aggregated-vlan
```

# alias

An alias serves as a shorthand version of a longer CLI command.

## Syntax

**alias**

**alias** *alias-name* **=** *cli-command*

**no alias** *alias-name*

**unalias** *alias-name*

## Command Default

No aliases are defined.

## Parameters

*alias-name*
: Alias name. Must be a single word, without spaces.

**=**
: Operator representing "equals."

*cli-command*
: Command string for which the alias is created.

## Modes

Privileged EXEC mode.

Global configuration mode.

## Usage Guidelines

To remove an alias you can enter the **no alias** or the **unalias** command followed by the *alias-name*.

An alias saves typing in a longer command that you commonly use. For example, you can create an alias called *shoro* for the CLI command **show ip route**. Then when you enter *shoro* at the command prompt, the **show ip route** command is issued.

Entering the **alias** command with no parameters displays the currently configured aliases on the device.

## Examples

The following example creates an alias called *shoro* for the CLI command **show ip route**, enter the **alias shoro = show ip route** command:

```
device(config)# alias shoro = show ip route
```

The following example uses the command **copy running-config** with the appropriate parameters to create an alias called *wrsbc*:

```
device(config)# alias wrsbc = copy running-config tftp 10.10.10.10 test.cfg
```

The following example removes the *wrsbc* alias from the configuration:

```
device(config)# no alias wrsbc
```

An alternate method of removing the alias is shown below:

```
device(config)# unalias wrsbc
```

To display the aliases currently configured on the Ruckus device, enter the following command at either the Privileged EXEC or global configuration modes of the CLI.

```
device# alias
          wrsbc      copy running-config tftp 10.10.10.10 test.cfg
          shoro       show ip route
```

# all-client

Restricts all remote management access methods to a host.

## Syntax

**all-client** { *ip-address* | **ipv6** *ipv6-address* }

**no all-client** { *ip-address* | **ipv6** *ipv6-address* }

## Command Default

Remote management access is not restricted.

## Parameters

*ip-address*
The IP address of the host to which you want to restrict the remote management access.

**ipv6** *ipv6-address*
The IPv6 address of the host to which you want to restrict the remote management access.

## Modes

Global configuration mode

## Usage Guidelines

By default, an ICX device does not control remote management access based on the IP address of the managing device. Using the **all-client** command, you can restrict remote management access to a single IP address for all of the following access methods:

- Telnet access
- SSH access
- Web management access
- SNMP access

You can specify only one IP address at a time. However, you can enter each command ten times to specify up to ten IP addresses.

The **no** form of the command removes the access restriction.

## Examples

The following example shows how to restrict all remote management access methods to a host with IP address 10.157.22.69.

```
device(config)# all-client 10.157.22.69
```

# always-compare-med

Configures the device always to compare the Multi-Exit Discriminators (MEDs), regardless of the autonomous system (AS) information in the paths.

## Syntax

**always-compare-med**

**no always-compare-med**

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command disallows the comparison of the MEDs for paths from neighbors in different autonomous systems.

## Examples

The following example configures the device always to compare the MEDs.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# always-compare-med
```

# always-propagate

Enables the device to reflect BGP routes even though they are not installed in the Routing Table Manager (RTM).

## Syntax

**always-propagate**

**no always-propagate**

## Command Default

This feature is disabled.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

This example configures the device to reflect routes that are not installed in the RTM.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# always-propagate
```

This example configures the device to reflect routes that are not installed in the RTM in IPv6 address-family unicast configuration mode.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# always-propagate
```

This example configures the device to reflect routes that are not installed in the RTM in a nondefault VRF instance.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# always-propagate
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# anycast-rp

Configures PIM anycast rendezvous points (RPs) in IPv4 and IPv6 multicast domains.

## Syntax

**anycast-rp** *rp-address anycast-rp-set-acl*

**no anycast-rp** *rp-address anycast-rp-set-acl*

## Command Default

PIM anycast RPs are not configured.

## Parameters

*rp-address*

Specifies a shared RP address used among multiple PIM routers.

*anycast-rp-set-acl*

Specifies a host-based simple access -control list (ACL) used to specify the address of the anycast RP set, including a local address.

## Modes

PIM router configuration mode

## Usage Guidelines

PIM anycast RP is a way to provide load balancing and fast convergence to PIM RPs in an IPv4 or IPv6 multicast domain. The RP address of the anycast RP is a shared address used among multiple PIM routers, known as PIM RP.

The PIM software supports up to eight PIM anycast RP routers. All deny statements in the my-anycast-rp-set-acl ACL are ignored.

The **no** form of the command removes the anycast RP configuration.

## Examples

The following example shows how to configure a PIM anycast RP.

```
device# configure terminal
device(config)# router pim
device(config-pim-router)# rp-address 100.1.1.1
device(config-pim-router)# anycast-rp 100.1.1.1 my-anycast-rp-set-acl
```

The following example shows how to configure PIM anycast RP 100.1.1.1.The example avoids using loopback 1 interface when configuring PIM Anycast RP because the loopback 1 address could be used as a router-id. A PIM first-hop router registers the source with the closest RP. The first RP that receives the register re-encapsulates the register to all other anycast RP peers.

```
device# configure terminal
device(config)# interface loopback 2
device(config-lbif-2)# ip address 100.1.1.1/24
device(config-lbif-2)# ip pim-sparse
device(config-lbif-2)# interface loopback 3
device(config-lbif-3)# ip address 1.1.1.1/24
device(config-lbif-3)# ip pim-sparse
device(config-lbif-3)# router pim
device(config-pim-router)# rp-address 100.1.1.1
device(config-pim-router)# anycast-rp 100.1.1.1 my-anycast-rp-set
device(config-pim-router)# ip access-list standard my-anycast-rp-set
device(config-std-nacl)# permit host 1.1.1.1
device(config-std-nacl)# permit host 2.2.2.2
device(config-std-nacl)# permit host 3.3.3.
```

The following example shows how to configure a PIM anycast RP for a VRF.

```
device# configure terminal
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# rp-address 1001::1
device(config-ipv6-pim-router-vrf-blue)# anycast-rp 1001::1 my-anycast-rp-set-acl
```

The following example shows how to configure PIM anycast RP 1001:1 so that it avoids using loopback 1.

```
device# configure terminal
device(config)# interface loopback 2
device(config-lbif-2)# ipv6 address 1001::1/96
device(config-lbif-2)# ipv6 pim-sparse
device(config-lbif-2)# interface loopback 3
device(config-lbif-3)# ipv6 address 1:1:1::1/96
device(config-lbif-3)# ipv6 pim-sparse
device(config-lbif-3)# ipv6 router pim
device(config-ipv6-pim-router)# rp-address 1001::1
device(config-ipv6-pim-router)# anycast-rp 1001::1 my-anycast-rp-set
device(config-ipv6-pim-router)# ipv6 access-list my-anycast-rp-set
device(config-std-nacl)# permit ipv6 host 1:1:1::1 any
device(config-std-nacl)# permit ipv6 host 2:2:2::2 any
device(config-std-nacl)# permit ipv6 host 3:3:3::3 any
```

# area authentication (IPsec)

Enables IPSec authentication for an OSPF Version 3 (OSPFv3) area.

## Syntax

**area** { *ip-address* | *decimal* } **authentication ipsec spi** *value* **esp sha1** *key*

**area** { *ip-address* | *decimal* } **authentication ipsec spi** *value* **esp sha1 no-encrypt** *key*

**no area** { *ipv6-address* | *decimal* } **authentication ipsec spi** *value*

## Command Default

Authentication is not enabled on an area.

The key is stored in encrypted format by default.

## Parameters

*ip-address*
Area ID in IP address format.

*decimal*
Area ID in decimal format.

**ipsec**
Specifies that IP security (IPsec) is the protocol that authenticates the packets.

**spi**
Specifies the Security Policy Index (SPI).

*value*
Specifies the SPI value. Valid values range from decimal numbers 256 through 4294967295. The near-end and far-end values must be the same.

**esp**
Specifies Encapsulating Security Payload (ESP) as the protocol to provide packet-level security. This is the only option currently available.

**sha1**
Enables Hashed Message Authentication Code (HMAC) Secure Hash Algorithm 1 (SHA-1) authentication on the OSPFv3 area.

*key*
Number used in the calculation of the message digest. The 40 hexadecimal character key is stored in encrypted format by default.

**no-encrypt**
The 40-character key is not encrypted upon either its entry or its display.

*key*
The 40 hexadecimal character key.

# Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

# Usage Guidelines

The 40 hexadecimal character key is encrypted by default. The system adds the following in the configuration to indicate that the key is encrypted:

- encrypt = the key string uses proprietary simple cryptographic 2-way algorithm
- encryptb64 = the key string uses proprietary base64 cryptographic 2-way algorithm

Use the **no-encrypt** parameter to disable encryption.

Currently certain keyword parameters must be entered though only one keyword choice is possible for that parameter. For example, the only authentication algorithm is HMAC-SHA1-96, but you must nevertheless enter the **sha1** keyword for this algorithm. Also, although ESP is currently the only authentication protocol, you must enter the **esp** keyword.

The **no** form of the command removes an authentication specification for an area from the configuration.

# Examples

The following example enables esp and SHA-1 authentication for an OSPFv3 area, setting a SPI value of 900.

```
device# configure terminal
device(config)# ip router-id 10.1.2.3
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 0 authentication ipsec spi 750 esp sha1
abcef12345678901234fedcba098765432109876
```

# area authentication (OSPFv3)

Configures HMAC-SHA-1 or HMAC-SHA-256 authentication for an Open Shortest Path First version 3 (OSPFv3) area.

## Syntax

**area** *area-id* **authentication** { **hmac-sha-1** | **hmac-sha-256**} **key-id** *key-id-val* **key** *key-string*

**no area** *area-id* **authentication** { **HMAC-SHA-1** | **HMAC-SHA-256**} **key-id** *key-id-val* **key** *key-string*

## Command Default

HMAC-SHA-1 or HMAC-SHA-256 authentication is disabled by default.

## Parameters

**hmac-sha-1**
Specifies the HMAC-SHA-1 authentication.

**hmac-sha-256**
Specifies the HMAC-SHA-256 authentication.

*key-id-val*
Identifies the number of the HMAC-SHA-1 or HMAC-SHA-256 algorithm. The number can be from 1 through 255.

*key-string*
Sets the corresponding key-string to be used with the HMAC-SHA-1 or HMAC-SHA-256 algorithm.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the HMAC-SHA-1 or HMAC-SHA-256 authentication configuration for an OSPFv3 area. All interfaces within the area are configured to use these authentication parameters.

It is possible to remove this configuration from individual interfaces using the **ipv6 ospf authentication disable** command on the required interface.

The **no** form of the command removes the HMAC-SHA-1 or HMAC-SHA-256 authentication configuration from the OSPFv3 area.

# Examples

The following example sets HMAC-SHA-1 authentication with key ID 10 and the password key "mypasswordkey", on the OSPFv3 area.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 authentication hmac-sha-1 key-id 10 key mypasswordkey
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area authentication keychain (OSPFv3)

Configures keychain authentication for an Open Shortest Path First version 3 (OSPFv3) area.

## Syntax

**area** *area-id* **authentication keychain** *keychain-name*

**no area** *area-id* **authentication keychain** *keychain-name*

## Command Default

Keychain authentication is disabled by default.

## Parameters

*keychain-name*
Specifies the name of the keychain that OSPFv3 uses to authenticate the packets.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the keychain authentication configuration for an OSPFv3 area. All interfaces within the area are configured to use these authentication parameters.

It is possible to remove this configuration from individual interfaces using the **ipv6 ospf authentication disable** command on the required interface.

The **no** form of the command removes the keychain authentication configuration from the OSPFv3 area.

## Examples

The following example configures the OSPFv3 area to use use the keychain authentication module with the "xtreme" keychain.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 authentication keychain xtreme
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area nssa (OSPFv2)

Creates a not-so-stubby area (NSSA) or modifies its parameters.

## Syntax

**area** { *ip-addr* | *decimal* } **nssa** { *metric* [ **no-summary** ] | **default-information-originate** }

**no area nssa**

## Command Default

No areas are created.

## Parameters

*ip-addr*
: Area address in IP address format.

*decimal*
: Area address In decimal format.

*metric*
: Additional cost for using a route to or from this area.

**no-summary**
: When configured on the NSSA area border router (ABR), this parameter prevents any Type 3 and Type 4 summary link-state advertisement (LSA) from being injected into the area. The only exception is that a default route is injected into the NSSA by the ABR, and strictly as a Type 3 LSA (not a Type 7, because that could cause intra-AS traffic to get routed out the AS). This makes the NSSA an NSSA totally stubby area, which can only have Type 1, 2 and 7 LSAs. **Note:** This parameter is disabled by default, which means the default route must use a Type 7 LSA.

**default-information-originate**
: When configured on the ABR, this parameter injects a Type 7 default route into the NSSA area. As a result, the other NSSA routers install the default route through the advertising NSSA ABR. By default the NSSA ABR does not originate a default route to the NSSA.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

NSSAs are typically needed when one-way transmission of Type-5 LSAs (out of the area) is desired but injection of the same LSAs into the area is not acceptable.

Once created, the type of the area cannot be changed. The only exception to this rule is that an NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a NSSA.

# Examples

The following example sets an additional cost of 5 on an NSSA identified as 2, includes the no-summary parameter, and prevents the device from importing type 3 and type 4 summary LSAs into the NSSA area.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# area 2 nssa 5 no-summary
```

# area nssa (OSPFv3)

Creates a not-so-stubby area (NSSA) or modifies its parameters.

## Syntax

**area** { *ip-addr* | *decimal* } **nssa** [ *metric* ] [ **default-information-originate** [ **metric** *num* ] [ **metric-type** { **type1** | **type2** } ] ] [ **no-redsitribution** ] [ **no-summary** ] [ **translator-always** ] [ **translator-interval** *interval* ]

**no area nssa**

## Command Default

No areas are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*metric*

Additional cost for using a route to or from this area. Valid values range from 1 through 1048575.

**default-information-originate**

When configured on the ABR, this parameter injects a Type 7 default route into the NSSA area. As a result, the other NSSA routers install the default route through the advertising NSSA ABR. By default the NSSA ABR does not originate a default route to the NSSA.

**metric-type**

Specifies how the cost of a neighbor metric is determined.

**type1**

The metric of a neighbor is the cost between itself and the router plus the cost of using this router for routing to the rest of the world.

**type2**

The metric of a neighbor is the total cost from the redistributing routing to the rest of the world.

**no-redistribution**

The no-redistribution parameter prevents an NSSA ABR from generating external (type-7) LSA into a NSSA area. This is used in the case where an ASBR should generate type-5 LSA into normal areas and should not generate type-7 LSA into a NSSA area. By default, redistribution is enabled in a NSSA.

**no-summary**

When configured on the NSSA area border router (ABR), this parameter prevents any Type 3 and Type 4 summary link-state advertisement (LSA) from being injected into the area. The only exception is that a default route is injected into the NSSA by the ABR, and strictly as a Type 3 LSA (not a Type 7, because that could cause intra-AS traffic to get routed out the AS). This makes the NSSA a NSSA totally stubby area, which can only have Type 1, 2 and 7 LSAs. **Note:** This parameter is disabled by default, which means the default route must use a Type 7 LSA.

**translator-always**

Configures the translator-role. When configured on an ABR, this causes the router to unconditionally assume the role of a NSSA translator. By default, translator-always is not set, the translator role by default is candidate.

**translator-interval** *interval*

Configures the time interval for which an elected NSSA translator continues to perform its duties even after its NSSA translator role has been disposed by another router. Valid values range from 10 through 60 seconds. By default the stability-interval is 40 seconds.

# Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

# Usage Guidelines

NSSAs are typically needed when one-way transmission of Type-5 LSAs (out of the area) is desired but injection of the same LSAs into the area is not acceptable.

Once created, the type of the area cannot be changed. The only exception to this rule is that a NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a NSSA.

# Examples

The following example sets an additional cost of 4 on a NSSA identified as 8 (in decimal format), and prevents any Type 3 or Type 4 summary LSAs from being injected into the area.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 8 nssa 4 no-summary
```

# area range (OSPFv2)

Specifies area range parameters on an area border router (ABR).

## Syntax

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L*

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L* **advertise** [ **cost** *cost_value* ]

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L* **cost** *cost_value*

**area** { *A.B.C.D* | *decimal* } **range** *E.F.G.H I.J.K.L* **not-advertise** [ **cost** *cost_value* ]

**no area range**

## Command Default

The address range is advertised.

## Parameters

*A.B.C.D*
Area address in IP address format.

*decimal*
Area address in decimal format.

*E.F.G.H I.J.K.L*
Specifies the IP address and mask portion of the range. All network addresses that match this network are summarized in a single route and advertised by the ABR.

**advertise**
Sets the address range status to *advertise* and generates a Type 3 summary LSA.

**cost** *cost_value*
Sets the cost value for the area range. This value is used as the generated summary LSA cost. The range for *cost_value* is 1 to 6777214. If this value is not specified, the cost value is the default range metric calculation for the generated summary LSA cost.

**not-advertise**
Sets the address range status to DoNotAdvertise; the Type 3 LSA is suppressed, and the component networks remain hidden from other networks. This setting is used to temporarily pause route summarization from the area.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

# Usage Guidelines

Use this command only on ABRs to specify route summarization for an existing area. The result is that a single summary route is advertised to other areas by the ABR, in the form of a Type 3 LSA. Routing information is condensed at area boundaries and external to the area, and only a single route is advertised for each address range.

An example of when you might want to use this command is if you have many small networks advertised from area 0 to any other area, or from any non-backbone area into the backbone. This command gives you a summary route instead of many smaller routes. In an area, the OSPF database on each router must be an exact copy of the databases of the other routers. This means that no summarization is allowed within the area.

The **no** form of the command disables the specification of range parameters on an ABR.

# Examples

The following example advertises to Area 3 all the addresses on the network 10.1.1.0 10.255.255.0 in the ABR you are signed into.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# area 3 range 10.1.1.0 10.255.255.0 advertise
```

# area range (OSPFv3)

Specifies area range parameters on an area border router (ABR).

## Syntax

**area** { *ip-addr* | *decimal* } **range** *ipv6 address/mask* [ **advertise** | **not-advertise** ] [ **cost** *cost_value* ]

**no area range**

## Parameters

*ip-addr*
Area address in IP address format.

*decimal*
Area address in decimal format.

*ipv6 address/mask*
Specifies the IPv6 address in dotted-decimal notation and the IPv6 mask in CIDR notation. All network addresses that match this network are summarized in a single route and advertised by the ABR.

**advertise**
Sets the address range status to *advertise* and generates a Type 3 summary LSA.

**cost** *cost_value*
Sets the cost value for the area range. This value is used as the generated summary LSA cost. The range for *cost_value* is 1 to 6777214. If this value is not specified, the cost value is the default range metric calculation for the generated summary LSA cost.

**not-advertise**
Sets the address range status to DoNotAdvertise; the Type 3 LSA is suppressed, and the component networks remain hidden from other networks. This setting is used to temporarily pause route summarization from the area.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command only on ABRs to specify route summarization for an existing area. The result is that a single summary route is advertised to other areas by the ABR, in the form of a Type 3 LSA. Routing information is condensed at area boundaries and external to the area, and only a single route is advertised for each address range.

An example of when you might want to use this command is if you have many small networks advertised from area 0 to any other area, or from any non-backbone area into the backbone. This command gives you a summary route instead of many smaller routes. In an area, the OSPF database on each router must be an exact copy of the databases of the other routers. This means that no summarization is allowed within the area.

The **no** form of the command disables the specification of range parameters on an ABR.

# Examples

The following example advertises to Area 3 all the addresses on the network 2001:db8:8::/45 in the ABR you are signed into.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 3 range 2001:db8:8::/45 advertise
```

# area stub (OSPFv2)

Creates or deletes a stub area or modifies its parameters.

## Syntax

**area** { *ip-addr* | *decimal* } **stub** *metric* [ **no-summary** ]

**no area stub**

## Command Default

No areas are created.

## Parameters

*A.B.C.D*

Area address in IP address format.

*decimal*

Area address In decimal format.

*metric*

Additional cost for using a route to or from this area. Valid values range from 1 through 6777215.

**no-summary**

When configured on the ABR, this parameter prevents any Type 3 and Type 4 summary LSAs from being injected into the area. The only exception is that a default route is injected into the stub/totally stubby area by the ABR as a Type 3 LSA. Enabling this parameter makes the area a so-called totally stubby area, which can only have Types 1 and 2. This parameter is disabled by default.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Once created, the type of the area cannot be changed. The only exception to this rule is that a NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a stub area.

## Examples

The following example sets an additional cost of 5 on a stub area called 2.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# area 2 stub 5
```

# area stub (OSPFv3)

Creates or deletes a stub area or modifies its parameters.

## Syntax

**area** { *ip-addr* | *decimal* } **stub** *metric* [ **no-summary** ]

**no area stub**

## Command Default

No areas are created.

## Parameters

*ip-addr*
Area address in IP address format.

*decimal*
Area address In decimal format.

*metric*
Additional cost for using a route to or from this area. Valid values range from 3 through 1048575.

**no-summary**
When configured on the ABR, this parameter prevents any Type 3 and Type 4 summary LSAs from being injected into the area. The only exception is that a default route is injected into the stub/totally stubby area by the ABR as a Type 3 LSA. Enabling this parameter makes the area a so-called totally stubby area, which can only have Types 1 and 2. This parameter is disabled by default.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Once created, the type of the area cannot be changed. The only exception to this rule is that a NSSA or stub area can be changed to a totally NSSA or a totally stub area, respectively.

The **no** form of the command deletes a stub area.

## Examples

The following example sets an additional cost of 5 on a stub area called 2.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 2 stub 5
```

# area virtual-link (OSPFv2)

Creates or modifies virtual links for an Open Shortest Path First version 2 (OSPFv2) area.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* [ **dead-interval** *time* ] [ **hello-interval** *time* ] [ **retransmit-interval** *time* ] [ **transmit-delay** *time* ]

**no area virtual-link**

## Command Default

No virtual links are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*E.F.G.H*

ID of the OSPF router at the remote end of the virtual link.

**dead-interval** *time*

How long a neighbor router waits for a hello packet from the current router before declaring the router down. This value must be the same for all routers and access servers that are attached to a common network. Valid values range from 3 through 65535 seconds. The default is 40 seconds.

**hello-interval** *time*

Time between hello packets that the router sends on an interface. The value must be the same for all routers and access servers that are attached to a common network. Valid values range from 1 through 65535 seconds. The default is 10 seconds.

**retransmit-interval** *time*

Time between Link State Advertisement (LSA) retransmissions for adjacencies belonging to the interface. Set this interval to a value larger than the expected round-trip delay between any two routers on the attached network. Valid values range from 0 through 3600 seconds. The default is 5 seconds.

**transmit-delay** *time*

Estimated time required to send an LSA on the interface. This value must be an integer greater than zero. The age of each LSA in the update packet is incremented by the value of this parameter before transmission occurs. Valid values range from 0 through 3600 seconds. The default is 1 second.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

The **no** form of the command removes a virtual link.

## Examples

The following example creates a virtual link for an area whose decimal address is 1, and where the ID of the OSPFv2 device at the remote end of the virtual link is 10.1.2.3.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# area 1 virtual-link 10.1.2.3
```

# area virtual-link (OSPFv3)

Creates or modifies virtual links for an area.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *A.B.C.D* [ **dead-interval** *time* | **hello-interval** *time* | **hello-jitter** *interval* | **retransmit-interval** *time* | **transmit-delay** *time* ]

**no area virtual-link**

## Command Default

No virtual links are created.

## Parameters

*ip-addr*

Area address in IP address format.

*decimal*

Area address in decimal format.

*A.B.C.D*

ID of the OSPFv3 device at the remote end of the virtual link.

**dead-interval** *time*

How long a neighbor device waits for a hello packet from the current device before declaring the device down. This value must be the same for all devices and access servers that are attached to a common network. Valid values range from 3 through 65535 seconds. The default is 40 seconds.

**hello-interval** *time*

Time between hello packets that the device sends on an interface. The value must be the same for all devices and access servers that are attached to a common network. Valid values range from 1 through 65535 seconds. The default is 10 seconds.

**hello-jitter** *interval*

Sets the allowed jitter between hello packets. Valid values range from 1 through 50 percent (%). The default value is 10%.

**retransmit-interval** *time*

Time between Link State Advertisement (LSA) retransmissions for adjacencies belonging to the interface. Set this interval to a value larger than the expected round-trip delay between any two devices on the attached network. Valid values range from 0 through 3600 seconds. The default is 5 seconds.

**transmit-delay** *time*

Estimated time required to send an LSA on the interface. This value must be an integer greater than zero. The age of each LSA in the update packet is incremented by the value of this parameter before transmission occurs. Valid values range from 0 through 3600 seconds. The default is 1 second.

# Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

# Usage Guidelines

The values of the **dead-interval** and **hello-interval** parameters must be the same at both ends of a virtual link. Therefore, if you modify the values of these parameters at one end of a virtual link, you must make the same modifications on the other end of the link. The values of the other virtual link parameters do not require synchronization.

The **no** form of the command removes a virtual link.

# Examples

The following example creates a virtual link for an area whose decimal address is 1, and where the ID of the OSPFv3 device at the remote end of the virtual link is 209.157.22.1.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 virtual-link 209.157.22.1
```

# area virtual-link authentication (OSPFv2)

Configures MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication for an Open Shortest Path First version 2 (OSPFv2) area virtual link.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication** { **md5** | **hmac-sha-1** | **hmac-sha-256**} **key-id** *key-id-val* **key** *key-string*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication**

## Command Default

MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication is disabled by default.

## Parameters

*ip-addr*
Area address in IP address format.

*decimal*
Area address in decimal format.

*E.F.G.H*
ID of the OSPF router at the remote end of the virtual link.

*md5*
Specifies MD5 authentication.

*HMAC-SHA-1*
Specifies HMAC-SHA-1 authentication.

*HMAC-SHA-256*
Specifies HMAC-SHA-256 authentication.

*key-id-val*
Identifies the number of the MD5, HMAC-SHA-1 or HMAC-SHA-256 algorithm. The number can be from 1 through 255.

*key-string*
Sets the corresponding key string to be used with the MD5, HMAC-SHA-1 or HMAC-SHA-256 algorithm. The recommended key string length is 1 through 63 characters.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

# Usage Guidelines

Use this command to set or reset the MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication configuration on the OSPFv2 area virtual link.

The **no** form of the command removes MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication configuration from the OSPFv2 area virtual link.

# Examples

The following example enables HMAC-SHA-1 authentication using the key ID 10 and key string "mypasswordkey" on the specified area virtual link.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# router ospf
device(config-ospf-router)# area 1 virtual-link 20.1.1.1 authentication hmac-sha-1 key-id 10 key
mypasswordkey
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area virtual-link authentication (OSPFv3)

Enables HMAC-SHA-1 or HMAC-SHA-256 authentication for virtual links in an OSPFv3 area.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication** { **hmac-sha-1** | **hmac-sha-256**} **key-id** *key-id-val* **key** *key-string*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication**

## Command Default

Authentication is not enabled on a virtual-link.

## Parameters

*ip-addr*
: Area address in IP address format.

*decimal*
: Area address in decimal format.

*E.F.G.H*
: ID of the OSPF router at the remote end of the virtual link.

*HMAC-SHA-1*
: Specifies HMAC-SHA-1 authentication.

*HMAC-SHA-256*
: Specifies HMAC-SHA-256 authentication.

*key-id-val*
: Identifies the number of the HMAC-SHA-1 or HMAC-SHA-256 algorithm. The number can be from 1 through 255.

*key-string*
: Sets the corresponding key string to be used with the HMAC-SHA-1 or HMAC-SHA-256 algorithm. The recommended key string length is 1 through 63 characters.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command to set or reset the HMAC-SHA-1 or HMAC-SHA-256 authentication configuration on the OSPFv3 area virtual link.

The **no** form of the command removes HMAC-SHA-1 or HMAC-SHA-256 authentication configuration from the OSPFv3 area virtual link.

# Examples

The following example enables HMAC-SHA-1 authentication using the key ID 10 and key string "mypasswordkey" on the specified area virtual link.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 virtual-link 20.1.1.1 authentication hmac-sha-1 key-id 10 key
mypasswordkey
```

# area virtual-link authentication ipsec (OSPFv3)

Enables IPsec (IP Security) authentication for virtual links in an OSPFv3 area.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication ipsec spi** *value* **esp sha1** *key* [ **no-encrypt** ] *key*

**no area** { *IPv6 address* | *decimal* } **virtual-link** *E.F.G.H* **authentication ipsec spi** *spi*

## Command Default

Authentication is not enabled on a virtual-link.

The 40 hexadecimal character key is encrypted by default. Use the **no-encrypt** parameter to disable encryption.

## Parameters

*ip-addr*
Area address in IP address format.

*decimal*
Area address in decimal format.

*E.F.G.H*
ID of the OSPFv3 device at the remote end of the virtual link.

**ipsec**
Specifies that IP security (IPsec) is the protocol that authenticates the packets.

**spi**
Specifies the Security Policy Index (SPI).

*value*
Specifies the SPI value. Valid values range from decimal numbers 256 through 4294967295. The near-end and far-end values must be the same.

**esp**
Specifies Encapsulating Security Payload (ESP) as the protocol to provide packet-level security. This is the only option currently available.

**sha1**
Enables Hashed Message Authentication Code (HMAC) Secure Hash Algorithm 1 (SHA-1) authentication on the OSPFv3 area.

*key*
Number used in the calculation of the message digest. The 40 hexadecimal character key is stored in encrypted format by default.

**no-encrypt**
The 40-character key is not encrypted upon either its entry or its display.

*key*
The 40 hexadecimal character key.

# Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

# Usage Guidelines

Currently certain keyword parameters must be entered though only one keyword choice is possible for that parameter. For example, the only authentication algorithm is HMAC-SHA1-96, but you must nevertheless enter the **sha1** keyword for this algorithm. Also, although ESP is currently the only authentication protocol, you must enter the **esp** keyword.

The **no** form of the command removes authentication from the virtual-links in the area.

# Examples

The following example configures IPsec on a virtual link in an OSPFv3 area, and encryption is disabled.

```
device# configure terminal
device(config)# ip router-id 10.1.2.2
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 2 virtual-link 10.1.2.2 authentication ipsec spi 600 esp sha1 no-
encrypt 113456789022345678901234567890123456789O
```

# area virtual-link authentication key-activation-wait-time (OSPFv2)

Configures the time before an authentication key change is activated for an Open Shortest Path First version 2 (OSPFv2) area virtual link.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication key-activation-wait-time** *wait-time*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication key-activation-wait-time** *wait-time*

## Command Default

The keychain wait time default is 300 seconds.

## Parameters

*ip-addr*
: Area address in IP address format.

*decimal*
: Area address in decimal format.

*E.F.G.H*
: ID of the OSPF router at the remote end of the virtual link.

*wait-time*
: Specifies the time before an authentication key change takes place. The wait time can be set from 0 through 14400 seconds.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use this command to set or reset the wait time before an authentication key change takes place on the OSPFv2 area virtual link.

The **no** form of the command resets the wait time to the default of 300 seconds.

# Examples

The following example sets the wait time before an authentication key change to 600 seconds on the OSPFv2 area virtual link.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# router ospf
device(config-ospf-router)# area 1 virtual-link 20.1.1.1 authentication key-activation-wait-time 600
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area virtual-link authentication key-activation-wait-time (OSPFv3)

Configures the time before an authentication key change is activated for an Open Shortest Path First version 3 (OSPFv3) area virtual link.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication key-activation-wait-time** *wait-time*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication key-activation-wait-time** *wait-time*

## Command Default

The keychain wait time default is 300 seconds.

## Parameters

*ip-addr*
: Area address in IP address format.

*decimal*
: Area address in decimal format.

*E.F.G.H*
: ID of the OSPF router at the remote end of the virtual link.

*wait-time*
: Specifies the time before an authentication key change takes place. The wait time can be set from 0 through 14400 seconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command to set or reset the wait time before an authentication key change takes place on the OSPFv3 area virtual link.

The **no** form of the command resets the wait time to the default of 300 seconds.

# Examples

The following example sets the wait time before an authentication key change to 600 seconds on the OSPFv3 area virtual link.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 virtual-link 20.1.1.1 authentication key-activation-wait-time 600
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area virtual-link authentication keychain (OSPFv2)

Configures keychain authentication for Open Shortest Path First version 2 (OSPFv2) area virtual link.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication keychain** *keychain-name*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication**

## Command Default

Keychain authentication is disabled by default.

## Parameters

*ip-addr*
Area address in IP address format.

*decimal*
Area address in decimal format.

*E.F.G.H*
ID of the OSPF router at the remote end of the virtual link.

**authentication keychain** *keychain-name*
Specifies the name of the keychain that OSPFv2 uses to authenticate the packets.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

The keychain authentication module provides the OSPFv2 protocol the option to automatically change the key ID and cryptographic algorithm without manual intervention.

With this configuration, OSPFv2 requests the keychain authentication module for all active keys in the keychain and selects the keys for sending and accepting the packets.

The **no** form of the command removes keychain authentication from the OSPFv2 area virtual link.

# Examples

The following example configures the OSPFv2 area virtual link to use the keychain authentication module with the "ruckus" keychain.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# router ospf
device(config-ospf-router)# area 1 virtual-link 20.1.1.1 authentication keychain ruckus
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area virtual-link authentication keychain (OSPFv3)

Configures keychain authentication for Open Shortest Path First version 3 (OSPFv3) area virtual link.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication keychain** *keychain-name*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication**

## Command Default

Keychain authentication is disabled by default.

## Parameters

*ip-addr*
: Area address in IP address format.

*decimal*
: Area address in decimal format.

*E.F.G.H*
: ID of the OSPF router at the remote end of the virtual link.

**authentication keychain** *keychain-name*
: Specifies the name of the keychain that OSPFv3 uses to authenticate the packets.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

The keychain authentication module provides the OSPFv3 protocol the option to automatically change the key ID and cryptographic algorithm without manual intervention.

With this configuration, OSPFv3 requests the keychain authentication module for all active keys in the keychain and selects the keys for sending and accepting the packets.

The **no** form of the command removes keychain authentication from the OSPFv3 area virtual link.

# Examples

The following example configures the OSPFv3 area virtual link to use the keychain authentication module with the "xtreme" keychain.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 virtual-link 20.1.1.1 authentication keychain ruckus
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area virtual-link authentication plain-text (OSPFv2)

Configures simple password-based authentication for an Open Shortest Path First version 2 (OSPFv2) area.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication plain-text** *key-string*

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication plain-text** *key-string*

## Command Default

Password-based authentication is disabled by default.

## Parameters

*ip-addr*
: Area address in IP address format.

*decimal*
: Area address in decimal format.

*E.F.G.H*
: ID of the OSPF router at the remote end of the virtual link.

*key-string*
: Sets the authentication password. The key string is unencrypted and appended to the outgoing message.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use this command to set or reset simple password-based authentication on the OSPFv2 area virtual link.

The **no** form of the command removes plain text authentication from the OSPFv2 area virtual link.

## Examples

The following example configures the authentication password "mystring" in plain text on the OSPFv2 area virtual link.

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# router ospf
device(config-ospf-router)# area 1 virtual-link 20.1.1.1 authentication plain-text mystring
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# area virtual-link authentication rfc6506 (OSPFv3)

Configures keychain authentication in accordance with RFC 6506 for an Open Shortest Path First version 3 (OSPFv3) area virtual link.

## Syntax

**area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication rfc6506**

**no area** { *ip-addr* | *decimal* } **virtual-link** *E.F.G.H* **authentication**

## Command Default

RFC 6506 authentication is disabled by default.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command to set or reset authentication in accordance with RFC 6506 on the OSPFv3 area virtual link. This may be required for backward compatibility. Although RFC 6506 is superseded by RFC 7166, some vendors continue to support RFC 6506. To ensure interoperability with vendor equipment that supports RFC 6506, use this command in conjunction with the required authentication options, as shown in the example below.

The **no** form of the command removes the RFC 6506 authentication configuration from the OSPFv3 area virtual link.

## Examples

The following example sets HMAC-SHA-1 authentication, in accordance with RFC 6506, on the OSPFv3 area virtual link. HMACSHA-1 authentication is enabled using key-id "1", key "0 1234567890123456789".

```
device# configure terminal
device(config)# ip router-id 10.1.1.1
device(config)# ipv6 router ospf
device(config-ospf6-router)# area 1 virtual-link 20.1.1.1 authentication rfc6506
device(config-ospf6-router)# area 1 virtual-link 20.1.1.1 authentication hmac-sha-1 key-id 1 key 0
1234567890123456789
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# arp

Creates static ARP entry.

## Syntax

**arp** *ip-address mac-address* { **ethernet** *unit/slot/port* | **lag** *lag-id* | **inspection** }

**no arp** *ip-address mac-address* { **ethernet** *unit/slot/portunit/slot/port* | **lag** *lag-id* | **inspection** }

## Command Default

Static ARP entries are not configured.

## Parameters

*ip-address*
Specifies the IP address of the device that has the MAC address of the entry.

*mac-address*
Specifies the MAC address of the entry.

**ethernet** *unit/slot/port*
Specifies the Ethernet interface.

**lag** *lag-id*
Specifies the LAG virtual interface.

**inspection**
Specifies the ARP inspection entry.

## Modes

Global configuration mode

## Usage Guidelines

Ruckus Layer 3 switches have a static ARP table, in addition to the regular ARP cache. The static ARP table contains entries that you configure.

Static entries are useful in cases where you want to pre-configure an entry for a device that is not connected to the Layer 3 switch, or you want to prevent a particular entry from aging out. The software removes a dynamic entry from the ARP cache if the ARP aging interval expires before the entry is refreshed. Static entries do not age out, regardless of whether the ICX device receives an ARP request from the device that has the entry address.

**NOTE**
You cannot create static ARP entries on a Layer 2 switch.

The maximum number of static ARP entries you can configure depends on the software version running on the device.

The **no** form of the command removes the configured static ARP entry.

# Examples

The following example creates a static ARP entry.

```
device(config)# arp 10.53.4.2 0000.0054.2348 ethernet 1/1/2
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# arp-internal-priority

Configures the priority of ingress ARP packets.

## Syntax

**arp-internal-priority** *priority-value*

## Command Default

The default priority of ingress ARP packets is 4.

## Parameters

*priority-value*

Specifies the priority value of the ingress ARP packets. It can take a value in the inclusive range of 0 to 7, where 7 is the highest priority.

## Modes

Global configuration mode

## Usage Guidelines

High traffic volume or non-ARP packets with a higher priority may cause ARP packets to be dropped, thus causing devices to become temporarily unreachable. You can use this command to increase the priority of ingress ARP packets. However, if the priority of ARP traffic is increased, a high volume of ARP traffic might cause drops in control traffic, possibly causing traffic loops in the network.

Stacking packets have a priority value of 7 and have higher precedence over ARP packets. If the ARP packets have priority value 7 in a stack system, they will be treated as priority value 6 packets when compared to stacking packets.

This command does not affect the priority of egress ARP packets.

You cannot change the priority of ingress ARP packets on the management port.

## Examples

The following example sets the priority of ingress ARP packets to a value of 7.

```
device(config)# arp-internal-priority 7
```

## History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced. |

# as-path-ignore

Disables the comparison of the autonomous system (AS) path lengths of otherwise equal paths.

## Syntax

**as-path-ignore**

**no as-path-ignore**

## Command Default

The comparison of the AS path lengths of otherwise equal paths is enabled.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command restores default behavior.

## Examples

The following example configures the device to always disable the comparison of AS path lengths.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# as-path-ignore
```

# atalk-proto

Configures the AppleTalk protocol-based VLAN.

## Syntax

**atalk-proto** [ **name** *string* ]

**no atalk-proto** [ **name** *string* ]

## Command Default

An AppleTalk protocol-based VLAN is not configured.

## Parameters

**name** *string*

Specifies the name of the AppleTalk protocol you want to configure on a VLAN. The name can be up to 32 characters in length.

## Modes

VLAN configuration mode

## Usage Guidelines

The device sends AppleTalk broadcasts to all ports within the AppleTalk protocol-based VLAN.

The **no** form of the command disables the AppleTalk protocol-based VLAN.

## Examples

The following example shows how to configure an AppleTalk protocol-based VLAN.

```
device(config)# vlan 10 by port
device(config-vlan-10)# untagged ethernet 1/1/1 to 1/1/6
added untagged port ethe 1/1/1 to 1/1/6 to port-vlan 30.
device(config-vlan-10)# atalk-proto name Atalk_Prot_VLAN
```

# attempt-max-num

Configures the number of times a user can enter an invalid username and password; that is, the number of Web Authentication attempts during the specified cycle time.

## Syntax

**attempt-max-num** *number*

**no attempt-max-num** *number*

## Command Default

The default number of Web Authentication attempts allowed is five.

## Parameters

*number*

Specifies the number of Web Authentication attempts. Valid values are from 0 through 64. If you configure 0, there is no limit on the number of attempts. The default is five attempts.

## Modes

Web Authentication configuration mode

## Usage Guidelines

You can set a limit on the number of times a user enters an invalid username and password during the specified cycle time. If the user exceeds the limit, the user is blocked for a duration of time, which is defined by the **block duration** command. Also, the Web browser will be redirected to the Exceeded Allowable Attempts web page.

The **no** form of the command sets the number of Web Authentication attempts to the default.

## Examples

The following example limits the number of Web Authentication attempts to 10.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# attempt-max-num 10
```

# auth allow-tagged enable

Allows tagged packets to be processed when the port is not tagged in the incoming tagged VLAN.

## Syntax

**auth allow-tagged enable**

**no auth allow-tagged enable**

## Command Default

By default, processing tagged packets is disabled.

## Modes

Interface configuration mode.

## Usage Guidelines

The **no** form of the command disables tagged packet processing.

## Examples

The following example enables tagged packet processing on port 1/1/1.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# auth allow-tagged enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# auth auth-mode

Configures Flexible authentication mode at the interface level.

## Syntax

**auth auth-mode** { **multiple-hosts** | **multiple-untagged** | **single-host** | **single-untagged** }

**no auth auth-mode** { **multiple-hosts** | **multiple-untagged** | **single-host** | **single-untagged** }

## Command Default

The default Flexible authentication mode at the interface level is single-untagged.

## Parameters

**multiple-hosts**
Configures the interface to operate in multiple-host authentication mode.

**multiple-untagged**
Configures the interface to operate in multiple-untagged authentication mode.

**single-host**
Configures the interface to operate in single-host authentication mode.

**single-untagged**
Configures the interface to operate in single-untagged authentication mode.

## Modes

Interface configuration mode.

## Usage Guidelines

The **auth auth-mode** configuration on a port overrides the global **auth-mode** configuration for the device.

The **no** form of the command returns the interface to the default authentication mode.

## Examples

The following example configures the Flexible authentication mode on port 1/1/1 as **multiple-hosts**.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# auth auth-mode multiple-hosts
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# auth-default-vlan

Specifies the auth-default VLAN globally.

## Syntax

**auth-default-vlan** *vlan-id*

**no auth-default-vlan** *vlan-id*

## Command Default

The auth-default VLAN is not specified.

## Parameters

*vlan-id*

Specifies the VLAN ID of the auth-default VLAN.

## Modes

Authentication configuration mode

## Usage Guidelines

The auth-default VLAN must be configured to enable authentication.

A VLAN must be configured as auth-default VLAN to enable authentication. When any port is enabled for 802.1X authentication or MAC authentication, the client is moved to this VLAN by default.

The auth-default VLAN is also used in the following scenarios:

- When the RADIUS server does not return VLAN information upon authentication, the client is authenticated and remains in the auth-default VLAN.
- If RADIUS timeout happens during the first authentication attempt and the timeout action is configured as "Success", the client is authenticated in the auth-default VLAN. If the RADIUS server is not available during reauthentication of a previously authenticated client, the client is retained in the previously authenticated VLAN.

The **no** form of the command disables the auth-default VLAN.

## Examples

The following example creates an auth-default VLAN with VLAN 2.

```
device(config)# authentication
device(config-authen)# auth-default-vlan 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# auth-fail-action (flexible authentication)

Configures, at a global level, the action taken after 802.1X and MAC authentication failure.

## Syntax

**auth-fail-action restricted-vlan** [ **voice voice-vlan** ]

**no auth-fail-action restricted-vlan** [ **voice voice-vlan** ]

## Command Default

The MAC address of the client is blocked in the hardware.

## Parameters

**restricted-vlan**

Places the client in the restricted VLAN after authentication failure.

**voice voice-vlan**

Places the client in the voice VLAN after authentication failure.

## Modes

Authentication configuration mode

## Usage Guidelines

**NOTE**
The **auth-fail-action** command takes effect only when flexible authentication is enabled on the ports. Therefore, flexible authentication must be enabled on ports prior to configuring the authentication failure action. The authentication failure action must also be reconfigured after a change to the flexible authentication status of a port.

Before setting the authentication failure action to **restricted-vlan**, the restricted VLAN must be configured using the **restricted-vlan** command.

The authentication failure action can be configured globally or at the interface level. When both global and interface-level authentication failure actions are configured, the interface-level configuration takes precedence. Authentication failure action is configured at interface level by using the **authentication fail-action** command.

In single untagged mode, client ports that are placed in the RADIUS-specified VLAN upon successful authentication are not placed in the restricted VLAN when subsequent authentication fails. Instead, the non-authenticated client is blocked.

When voice VLAN is configured, clients are placed in the voice VLAN as a tagged member.

The **no** form of the command removes the authentication failure action configuration.

# Examples

The following example configures using VLAN 4 as the restricted VLAN and then specifies placing the client in the restricted VLAN after authentication failure.

```
device(config)# authentication
device(config-authen)# restricted-vlan 4
device(config-authen)# auth-fail-action restricted-vlan
```

The following example specifies placing the client in the restricted VLAN and the voice VLAN after authentication failure.

```
device(config)# authentication
device(config-authen)# restricted-vlan 4
device(config-authen)# auth-fail-action restricted-vlan voice voice-vlan
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.61 | This command was modified to support configuration of an authentication failure action for voice traffic. |

# auth-mode

Specifies the authentication mode, for example, as single-host, multiple, host, or multiple-untagged.

## Syntax

**auth-mode** { **multiple-hosts** | **multiple-untagged** | **single-host** | **single-untagged** }

## Command Default

By default, multiple-untagged is the authentication mode.

## Parameters

**multiple-hosts**
Specifies that Flexible authentication operates in multiple host mode.

**multiple-untagged**
Specifies that Flexible authentication operates in multiple untagged mode.

**single-host**
Specifies that Flexible authentication operates in single host mode.

**single-untagged**
Configures the device to operate in single-untagged authentication mode.

## Usage Guidelines

The **auth auth-mode** configuration on a port overrides the global **auth-mode** configuration for the device.

## Modes

authentication configuration sub-mode.

## Examples

The following example configures Flexible authentication single host mode.

```
device# configure terminal
device(config)# authentication
device(config-authen)# single-host
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# auth-mode captive-portal

Authenticates the users in a VLAN through external Web Authentication (Captive Portal user authentication mode).

## Syntax

**auth-mode captive-portal**

**no auth-mode captive-portal**

## Command Default

External Web Authentication mode is not enabled by default.

## Modes

Web Authentication configuration mode

## Usage Guidelines

External Web Authentication uses RADIUS as the authentication method.

The **no** form of the command removes the external Web Authentication mode as the configured authentication mode.

## Examples

The following example configures the authentication mode as external Web Authentication to authenticate the users in a VLAN.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode captive-portal
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j |

# auth-mode none

Enables automatic Web Authentication.

## Syntax

**auth-mode none**

**no auth-mode none**

## Command Default

By default, if Web Authentication is enabled, hosts need to login and enter authentication credentials to gain access to the network.

## Modes

Web Authentication configuration mode

## Usage Guidelines

If a reauthentication period is configured, the host will be asked to re-enter authentication credentials once the reauthentication period ends.

You can configure Web Authentication to authenticate a host when the user clicks the **Login** button. When a host enters a valid URL address, Web Authentication checks the list of blocked MAC addresses. If the host's MAC address is not on the list and the number of allowable hosts has not been reached, after clicking the **Login** button, the host is automatically authenticated for the duration of the configured reauthentication period, if one is configured. Once the reauthentication period ends, the host is logged out and must enter the URL address again. If automatic authentication is enabled and a host address is not in the blocked MAC address list, Web Authentication authenticates the host and displays the Login page without user credentials, and then provides a hyperlink to the requested URL site.

**NOTE**
Automatic authentication is not the same as permanent authentication. You must still specify devices that are to be permanently authenticated even if automatic authentication is enabled.

Use the **show webauth vlan** command in VLAN configuration mode to determine if automatic authentication is enabled.

The **no** form of the command removes the automatic Web Authentication configuration.

## Examples

The following example enables automatic Web Authentication.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode none
```

# auth-mode passcode

Enables Web Authentication to use dynamically created passcodes to authenticate users in the VLAN.

## Syntax

**auth-mode passcode** [ **flush-expired** | **generate** | **grace-period** *time* | **length** *passcode-length* | **log** { **snmp-trap** | **syslog** } | **refresh-type** { **duration** *time* | **time** [ *time-string* | **delete-all** ] } | **resend-log** | **static** ]

**no auth-mode passcode** [ **flush-expired** | **generate** | **grace-period** *time* | **length** *passcode-length* | **log** { **snmp-trap** | **syslog** } | **refresh-type** { **duration** *time* | **time** [ *time-string* | **delete-all** ] } | **resend-log** | **static** ]

## Command Default

Passcode authentication is not enabled.

## Parameters

**flush-expired**
Deletes old passcodes that have expired but are still valid because they are in the grace period.

**generate**
Refreshes the passcode instead of waiting for the system to automatically generate one.

**grace-period** *time*
Configures a grace period for an expired passcode.

**length** *passcode-length*
Configures the passcode length. Valid values are from 4 through 16 digits. The default is 4 digits.

**log**
Enables the generation of syslog messages and SNMP trap messages every time a new passcode is generated and passcode authentication is attempted. By default, the syslog and SNMP trap messages are enabled.

**snmp-trap**
Generates SNMP trap messages every time a new passcode is generated and passcode authentication is attempted.

**syslog**
Generates syslog messages every time a new passcode is generated and passcode authentication is attempted.

**refresh-type**
Configures the passcode refresh type as one of the following:

**duration** *time*
Configures the duration of time after which passcodes are refreshed. By default, dynamically created passcodes are refreshed every 1440 minutes (24 hours).

**time** *time-string*
Configures the time of the day when the passcode should be refreshed. When initially enabled, the time of day method will cause passcodes to be refreshed at 00:00 (12:00 midnight). You can add up to 24 refresh periods in a 24-hour period.

**delete-all**

Deletes all of the configured passcode refresh times and reverts back to the default time of 00:00 (12:00 midnight).

**resend-log**

Retransmits the current passcode to a syslog message or SNMP trap if passcode logging is enabled.

**static**

Creates a static passcode.

# Modes

Web Authentication configuration mode

# Usage Guidelines

You can delete old passcodes that have expired but are still valid because they are in the grace period using the **auth-mode passcode flush-expired** command. This is useful in situations where the old passcodes have been compromised but are still valid because of the grace period. This command does not affect current valid passcodes or passcodes that newly expire.

When manually refreshed using the **auth-mode passcode generate** command, the old passcode will no longer work, even if a grace period is configured. Also, if the passcode refresh method duration of time is used, the duration counter is reset when the passcode is manually refreshed. The passcode refresh method time of day is not affected when the passcode is manually refreshed.

If the grace period is reconfigured using the **auth-mode passcode grace-period** command while a passcode is already in the grace period, the passcode is not affected by the configuration change. The new grace period will apply only to passcodes that expire after the new grace period is set.

If you change the passcode refresh value using the **auth-mode passcode refresh-type**, the configuration is immediately applied to the current passcode. If both the duration of time and time of day passcode refresh values are configured, they are saved to the configuration file. You can switch back and forth between the passcode refresh methods, but only one method can be enabled at a time.

Passcodes are not stateful, meaning a software reset or reload will cause the system to erase the passcode. When the device comes back up, a new passcode will be generated.

When the **auth-mode passcode resend-log** command is configured, the switch retransmits the current passcode only. Passcodes that are in the grace period are not sent.

Static passcodes can be used for troubleshooting purposes, or for networks that want to use passcode authentication, but do not have the ability to support automatically generated passcodes (for example, the network does not fully support the use of SNMP traps or syslog messages with passcodes). Manually created passcodes are used in conjunction with dynamic passcodes . You can configure up to four static passcodes that never expire. Unlike dynamically created passcodes, static passcodes are saved to flash memory. By default, there are no static passcodes configured on the switch. Static passcodes do not have to be the same length as passcodes that are automatically generated.

Use the **show webauth vlan** *vlan-id* **passcode** command to view the current passcodes.

The **no** form of the command removes or disables the configured settings.

# Examples

The following example flushes out all expired passcodes that are currently in the grace period.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode flush-expired
```

The following example refreshes the passcode immediately.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode generate
```

The following example configures the grace period for an expired passcode.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode grace-period 5
```

The following example increases the passcode length to 10 digits.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode length 10
```

The following example shows how to re-enable syslog messages for passcodes after they have been disabled.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode log syslog
```

The following example changes the duration of time after which passcodes are refreshed to 4320 minutes (72 hours).

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode refresh-type duration 4320
```

The following example configures the switch to refresh passcodes at a certain time of day.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode refresh-type time 14:30
```

The following example deletes all of the configured passcode refresh times and reverts back to the default time of 00:00 (12:00 midnight).

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode refresh-type time delete-all
```

The following example retransmits the current passcode to a syslog message or SNMP trap if passcode logging is enabled.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode resend-log
```

The following example creates static passcodes.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode passcode static 3267345
```

# auth-mode username-password

Enables the username and password Web Authentication mode.

## Syntax

**auth-mode username-password** [ **auth-methods** {**radius** [ **local** ] | **local** [**radius** ] } | **local-user-database** *database-name* ]

**no auth-mode username-password** [ **auth-methods** {**radius** [ **local** ] | **local** [**radius** ] } | **local-user-database** *database-name* ]

## Command Default

Username password authentication is not enabled.

## Parameters

**auth-methods**
Configures the authentication method.

**radius**
Uses the RADIUS server to authenticate.

**local**
Uses the local user database to authenticate.

**local-user-database** *database-name*
Uses the usernames and passwords in the specified database to authenticate.

## Modes

Web Authentication configuration mode

## Usage Guidelines

You can optionally specify a failover sequence for RADIUS and local user database authentication methods. For example, you can configure Web Authentication to first use a local user database to authenticate users in a VLAN. If the local user database is not available, it will use a RADIUS server. You can specify the **local** and **radius** options one after the other in the required sequence to configure the failover sequence.

The **no** form of the command removes the username password authentication.

## Examples

The following example uses a local user database to authenticate users in a VLAN.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode username-password auth-methods local
```

The following example uses the usernames and passwords in the specified database to authenticate.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode username-password local-user-database
```

The following example configures a failover sequence for RADIUS and local user database authentication methods. In this example, Web Authentication first uses a local user database to authenticate users in a VLAN. If the local user database is not available, it will use a RADIUS server.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# auth-mode username-password auth-methods local radius
```

# auth-order

Specifies the sequence of authentication methods, 802.1X authentication and MAC authentication at the global level.

## Syntax

**auth-order** {**dot1x mac-auth** | **mac-auth dot1x** }

**no auth-order** {**dot1x mac-auth** | **mac-auth dot1x** }

## Command Default

The authentication sequence is set to perform 802.1X authentication method followed by MAC authentication.

## Parameters

**dot1x mac-auth**

Specifies 802.1X authentication followed by MAC authentication as the order of authentication methods on the interface.

**mac-auth dot1x**

Specifies MAC authentication followed by 802.1X authentication as the order of authentication methods on the interface.

## Modes

Authentication configuration mode

## Usage Guidelines

If 802.1X authentication and MAC authentication methods are enabled on the same port, by default the authentication sequence is set to perform 802.1X authentication followed by MAC authentication.

For authentication order 802.1X authentication followed by MAC authentication: When 802.1X authentication succeeds, the client is authenticated and the policies returned by the RADIUS server are applied. MAC authentication is not performed in this case. If 802.1X authentication fails, the failure action is carried out and MAC authentication is not attempted. On the other hand, if the client does not respond to 802.1X messages, then MAC authentication is attempted. Upon successful MAC authentication, the client is authenticated and the policies returned by the RADIUS server are applied and on authentication failure, the configured failure action is applied.

For authentication order MAC authentication followed by 802.1X authentication: By default, 802.1X authentication is performed even if MAC authentication is successful. Upon successful 802.1X authentication, the client is authenticated and the policies returned by the RADIUS server are applied and on authentication failure, the configured failure action is applied. The default behavior can be changed by specifying the RADIUS attribute, to prevent the 802.1X authentication from being performed after successful MAC authentication. In this case, the client is authenticated and the policies returned by the RADIUS server are applied after successful MAC authentication. If MAC authentication method fails, 802.1X port security authentication is not attempted and the configured failure action is applied. However, if the **mac-authentication dot1x-override** command is configured, the clients that failed MAC authentication undergoes 802.1X

authentication if the failure action is configured as restricted VLAN. If 802.1X authentication is successful, the policies returned by the RADIUS server are applied to the port.

The **no** form of the command disables the authentication order functionality.

# Examples

The following example specifies 802.1X authentication followed by MAC authentication as the order of authentication methods at the global level.

```
device(config)# authentication
device(config-authen)# auth-order dot1x mac-auth
```

The following example specifies MAC authentication followed by 802.1X authentication as the order of authentication methods at the global level.

```
device(config)# authentication
device(config-authen)# auth-order mac-auth dot1x
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# auth-timeout-action

Configures, at a global level, the action taken when external server authentication times out.

## Syntax

**auth-timeout-action** { **critical-vlan** [ **voice voice-vlan** ] | **failure** | **success** }

**no auth-timeout-action** { **critical-vlan** [ **voice voice-vlan** ] | **failure** | **success** }

## Command Default

Authentication timeout action is not configured at a global level.

## Parameters

**critical-vlan**
Places the client in the critical VLAN after RADIUS timeout.

**voice voice-vlan**
Places the client in the voice VLAN after RADIUS timeout.

**failure**
Specifies that RADIUS timeout causes authentication failure.

**success**
Specifies that RADIUS timeout causes authentication success.

## Modes

Authentication configuration mode

## Usage Guidelines

**NOTE**
The **auth-timeout-action** command takes effect only when flexible authentication is enabled on the ports. Therefore, flexible authentication must be enabled on ports prior to configuring the RADIUS timeout action. The RADIUS timeout action must also be reconfigured after a change to the flexible authentication status of a port.

The **auth-timeout-action** command configures the RADIUS timeout action at a global level.

The **success** option triggers authentication success and the client is placed in the previously-authenticated VLAN. In the case of first time authentication, the client is placed in the default voice VLAN.

The **failure** option causes authentication failure and results in the execution of the authentication failure action. The authentication failure action is configured at a global level by using the **auth-fail-action** command and at the local interface level by using the **authentication** command.

RADIUS timeout action can also be configured at the port level by using the **authentication timeout-action** command. When authentication timeout actions are configured at both global and local port level, the port-level configuration takes precedence.

The **no** form of the command removes the authentication timeout action configuration.

# Examples

The following example specifies placing the client in the critical VLAN and the voice VLAN (for voice traffic) after RADIUS authentication timeout.

```
device# configure terminal
device(config)# authentication
device(config-authen)# auth-timeout-action critical-vlan voice voice-vlan
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# auth-vlan-mode

Enables the Flexible authentication-enabled ports to be member of multiple untagged VLANs.

## Syntax

**auth-vlan-mode** { **multiple-untagged** }

**no auth-vlan-mode** { **multiple-untagged** }

## Command Default

Flexible authentication-enabled port can be member of only one untagged VLAN.

## Parameters

**multiple-untagged**

Allows the client to be assigned to multiple untagged VLANs on authentication.

## Modes

Authentication configuration mode

## Usage Guidelines

Reload is not required to change the VLAN mode. If the command is applied globally, all sessions will be cleared on all interfaces that have Flexible authentication enabled. However, existing sessions will be cleared if the command is applied on an individual interface using the **authentication auth-vlan-mode** command from the interface configuration mode.

Single untagged mode is only applicable to untagged VLANs returned by RADIUS.

The **no** form of the command returns the VLAN mode to single untagged. Port can be assigned to only one untagged VLAN on authentication.

## Examples

The following example configures multiple untagged VLAN at the global level.

```
device# configure terminal
device(config)# authentication
device(config-authen)# auth-vlan-mode multiple-untagged
```

The following example clears all sessions on interfaces with Flexible authentication enabled and restores the single untagged VLAN mode default on all new sessions established on those interfaces.

```
device# configure terminal
device(config)# authentication
device(config-authen)# no auth-vlan-mode multiple-untagged
```

# History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was introduced. |

# authenticate

Enables Network Time Protocol (NTP) strict authentication.

## Syntax

**authenticate**

**no authenticate**

## Command Default

Authentication is disabled.

## Modes

NTP configuration mode

## Usage Guidelines

If authentication is enabled, NTP packets that do not have a valid MAC address are dropped.

The **no** form of the command disables NTP strict authentication.

## Examples

The following example enables NTP strict authentication.

```
device(config)# ntp
device(config-ntp)# authenticate
```

# authenticated-mac-age-time

Configures the time duration after which the user-associated MAC address is aged out and reauthentication is enforced.

## Syntax

**authenticated-mac-age-time** *time*

**no authenticated-mac-age-time** *time*

## Command Default

The default time is 3600 seconds.

## Parameters

*time*

Specifies the time duration after which the user-associated MAC address is aged out and reauthentication is enforced. Valid values are 0 seconds to the reauthentication time configured using the **reauth-time** command. The default value is 3600 seconds.

## Modes

Web Authentication configuration mode

## Usage Guidelines

You can force Web Authenticated hosts to be reauthenticated if they have been inactive for a period of time. The inactive duration is calculated by adding the **mac-age-time** that has been configured for the device and the configured **authenticated-mac-age-time**. The **mac-age-time** command defines how long a port address remains active in the address table. If the authenticated host is inactive for the sum of these two values, the host is forced to be reauthenticated.

The **no** form of the command sets the time to the default of 3600 seconds.

## Examples

The following example configures the time duration after which the user-associated MAC addressed is aged out and reauthentication is enforced.

```
device(config)# mac-age-time 600
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# reauth-time 303
device(config-vlan-10-webauth)# authenticated-mac-age-time 300
```

# authentication

Enters the authentication mode.

## Syntax

**authentication**

**no authentication**

## Command Default

Authentication mode is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command will disable the authentication functionality.

Use this command to enter the authentication mode from global configuration mode. After entering authentication mode, you can configure additional authentication functionality that applies globally. Authentication functionality is also available for configuration at the interface configuration mode using different commands that apply only to the specified interface.

## Examples

The following example enables authentication.

```
device(config)#authentication
device(config-authen)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication (IKEv2)

Configures an authentication proposal for an Internet Key Exchange version 2 (IKEv2) profile.

## Syntax

**authentication** *authentication-proposal-name*

**no authentication** *authentication-proposal-name*

## Command Default

The default authentication proposal is def-ike-auth-prop.

## Parameters

*authentication-proposal-name*
Specifies the name of an authentication proposal.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

When an IKEv2 profile is created, it uses the default authentication proposal (def-ike-auth-prop). The def-ike-auth-prop proposal has the following settings:

- Method for local device authentication: pre_shared
- Method for local device authentication: pre_shared
- Pre-shared key: $QG5HTT1Ebk1TVW5NLWIhVW5ATVMhLS0rc1VA

Use this command to configure an alternate authentication proposal for the IKEv2 profile.

The **no** form of the command restores the default configuration.

## Examples

The following example shows how to configure an authentication proposal named auth_test1 for an IKEv2 profile named ikev2_profile.

```
device# configure terminal
device(config)# ikev2 profile ikev2_profile
device(config-ikev2-profile-ikev2_profile)# authentication auth_test1
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# authentication auth-default-vlan

Specifies the authentication default VLAN at the interface level.

## Syntax

**authentication auth-default-vlan** *vlan-id*

**no authentication auth-default-vlan** *vlan-id*

## Command Default

The auth-default VLAN is not specified.

## Parameters

*vlan-id*

Specifies the VLAN ID of the auth-default VLAN.

## Modes

Interface configuration mode

## Usage Guidelines

The auth-default VLAN specified at the interface level overrides the auth-default VLAN configured using the **auth-default-vlan** command at the global level. The configured auth-default VLAN configured at the global level will still be applicable to other ports that don't have auth-default VLAN configured at the interface level.

The local auth-default VLAN must be configured to enable authentication.

A VLAN must be configured as auth-default VLAN to enable authentication. When any port is enabled for 802.1X authentication or MAC authentication, the client is moved to this VLAN by default.

The auth-default VLAN is also used in the following scenarios:

- When the RADIUS server does not return VLAN information upon authentication, the client is authenticated and remains in the auth-default VLAN.
- If RADIUS timeout happens during the first authentication attempt and the timeout action is configured as "Success", the client is authenticated in the auth-default VLAN. If the RADIUS server is not available during reauthentication of a previously authenticated client, the client is retained in the previously authenticated VLAN.

The **no** form of the command disables the auth-default VLAN.

## Examples

The following example creates a default VLAN with VLAN 3.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication auth-default-vlan 3
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication auth-order

Specifies the sequence of authentication methods, 802.1X authentication and MAC authentication, on a specific interface.

## Syntax

**authentication auth-order** {**dot1x mac-auth** | **mac-auth dot1x** }

**no authentication auth-order** {**dot1x mac-auth** | **mac-auth dot1x** }

## Command Default

The authentication sequence is set to perform 802.1X authentication method followed by MAC authentication.

## Parameters

**dot1x mac-auth**

Specifies 802.1X authentication followed by MAC authentication as the order of authentication methods on the interface.

**mac-auth dot1x**

Specifies MAC authentication followed by 802.1X authentication as the order of authentication methods on the interface.

## Modes

Interface configuration mode

## Usage Guidelines

If 802.1X authentication and MAC authentication methods are enabled on the same port, by default the authentication sequence is set to perform 802.1X authentication followed by MAC authentication.

Configuring the authentication order at the interface level overrides the configuration at the global level for that particular interface. The configured global authentication order will still be applicable to other ports that don't have a per port authentication order configured.

For authentication order 802.1X authentication followed by MAC authentication: When 802.1X authentication succeeds, the client is authenticated and the policies returned by the RADIUS server are applied. MAC authentication is not performed in this case. If 802.1X authentication fails, the failure action is carried out and MAC authentication is not attempted. On the other hand, if the client does not respond to dot1x messages, then MAC authentication is attempted. Upon successful MAC authentication, the client is authenticated and the policies returned by the RADIUS server are applied and on authentication failure, the configured failure action is applied.

For authentication order MAC authentication followed by 802.1X authentication: By default, 802.1X authentication is performed even if MAC authentication is successful. Upon successful 802.1X authentication, the client is authenticated and the policies returned by the RADIUS server are applied and on authentication failure, the configured failure action is applied. The default behavior can be changed by specifying the RADIUS attribute, to prevent the 802.1X authentication from being performed after successful MAC authentication. In this case, the client is authenticated and the policies

returned by the RADIUS server are applied after successful MAC authentication. If MAC authentication method fails, 802.1X port security authentication is not attempted and the configured failure action is applied. However, if the **mac-authentication dot1x-override** command is configured, the clients that failed MAC authentication undergoes 802.1X authentication if the failure action is configured as restricted VLAN. If 802.1X authentication is successful, the policies returned by the RADIUS server are applied to the port.

The **no** form of the command disables the authentication order functionality.

# Examples

The following example specifies 802.1X authentication followed by MAC authentication as the order of authentication methods on Ethernet interface 1/1/3.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/3
device(config-if-e1/1/3)# authentication auth-order dot1x mac-auth
```

The following example specifies MAC authentication followed by 802.1X authentication as the order of authentication methods on Ethernet interface 1/1/3.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/3
device(config-if-e1/1/3)# authentication auth-order mac-auth dot1x
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication auth-vlan-mode

Enables multiple-untagged mode on a specific Flexible authentication-enabled port and allows it to be member of multiple untagged VLANs.

## Syntax

**authentication auth-vlan-mode** { **multiple-untagged** }

**no authentication auth-vlan-mode** { **multiple-untagged** }

## Command Default

Flexible authentication-enabled port can be member of only one untagged VLAN.

## Parameters

**multiple-untagged**

Allows the client to be assigned to multiple untagged VLANs on authentication.

## Modes

Interface configuration mode

## Usage Guidelines

Reload is not required to change the VLAN mode. However, existing sessions will be cleared if the command is applied to an individual interface.

The VLAN mode specified at the interface level overrides the VLAN mode configured using the **auth-vlan-mode** command at the global level. The configured VLAN mode configured at the global level will still be applicable to other ports that don't have the VLAN mode configured at the interface level.

Single untagged mode is only applicable to untagged VLANs returned by RADIUS.

The **no** form of the command returns the VLAN mode to single untagged. Port can be assigned to only one untagged VLAN on authentication.

## Examples

The following example configures multiple untagged VLAN mode on interface 1/1/1.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
(config-if-e1000-1/1/1)# authentication auth-vlan-mode multiple-untagged
```

The following example clears all sessions on a Flexible authentication enabled interface and restores the single untagged VLAN mode.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
(config-if-e1000-1/1/1)# no authentication auth-vlan-mode multiple-untagged
```

# History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was introduced. |

# authentication disable-aging

Disables aging of MAC sessions at the interface level.

## Syntax

**authentication disable-aging** { **permitted-mac-only** | **denied-mac-only** }

**no authentication disable-aging** { **permitted-mac-only** | **denied-mac-only** }

## Command Default

Aging of MAC sessions is not disabled.

## Parameters

**permitted-mac-only**
Prevents permitted (authenticated and restricted) sessions from being aged out and ages denied sessions.

**denied-mac-only**
Prevents denied sessions from being aged out, but ages out permitted sessions.

## Modes

Interface configuration mode

## Usage Guidelines

Use this command to disable the aging of MAC sessions. Use the **authentication disable-aging** command at the interface level and the **disable-aging** command in the authentication configuration mode. Entered at the interface level, this command overrides the command entered at the authentication global level. However, the global configuration to disable aging of MAC sessions will still be applicable to other ports that don't have configuration at the interface level.

The **no** form of the command does not disable aging.

## Examples

The following example disables aging for permitted MAC addresses.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication disable-aging permitted-mac-only
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication dos-protection

Enables denial of service (DoS) authentication protection on the interface.

## Syntax

**authentication dos-protection** { **enable** | **mac-limit** *mac-limit-value* }

**no authentication dos-protection** { **enable** | **mac-limit** *mac-limit-value* }

## Command Default

Denial of service is disabled by default.

## Parameters

**enable**

Specifies to enable DoS protection.

**mac-limit**

Specifies the maximum number MAC-authentication attempts allowed per second.

*mac-limit-value*

Specifies the rate limit for DoS protection. You can specify a rate from 1 - 65535 authentication attempts per second. The default is a rate of 512 authentication attempts per second.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables DoS protection.

To limit the susceptibility of the ICX device to DoS attacks, you can configure the device to use multiple RADIUS servers, which can share the load when there are a large number of MAC addresses that need to be authenticated. The ICX device can run a maximum of 10 RADIUS clients per server and will attempt to authenticate with a new RADIUS server if current one times out.

In addition, you can configure the ICX device to limit the rate of authentication attempts sent to the RADIUS server. When MAC authentication is enabled, the number of RADIUS authentication attempts made per second is tracked. When you also enable the DoS protection feature, if the number of RADIUS authentication attempts for MAC addresses learned on an interface per second exceeds a configurable rate (by default 512 authentication attempts per second), the device considers this a possible DoS attack and disables the port. You must then manually re-enable the port.

# Examples

The example specifies the DoS protection count as 256.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# authentication dos-protection mac-limit 256
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication fail-action

Specifies the authentication failure action to move the client port to the restricted VLAN after authentication failure for both MAC authentication and 802.1X authentication on an interface.

## Syntax

**authentication fail-action restricted-vlan** *vlan-id*

**no authentication fail-action restricted-vlan**

## Command Default

The default action is to block the MAC address of the client.

## Parameters

**restricted-vlan**

Specifies the failure action to move the client port to the restricted VLAN after authentication failure.

*vlan-id*

Specifies the ID of the VLAN to be configured as restricted VLAN.

## Modes

Interface configuration mode

## Usage Guidelines

If the authentication failure action is not configured, the client's MAC address is blocked in the hardware (default action) when the authentication fails.

The restricted VLAN specified at the interface level overrides the restricted VLAN configured using the **restricted-vlan** command at the global level. The configured restricted VLAN configured at the global level will still be applicable to other ports that don't have restricted VLAN configured at the interface level.

The client ports that were placed in the RADIUS-specified VLAN upon successful authentication are not placed in the restricted VLAN if the subsequent authentication fails. Instead, the non-authenticated client is blocked.

The **no** form of the command disables the authentication failure action.

## Examples

The following example specifies authentication failure action to move the client port to the restricted VLAN (VLAN 4 is configured as restricted VLAN) after authentication failure.

```
device(config)# authentication
device(config-authen)# restricted-vlan 4
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication fail-action restricted-vlan 5
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication filter-strict-security

Enables or disables strict filter security for 802.1X and MAC-authentication enabled interfaces.

## Syntax

**authentication filter-strict-security**

**no authentication filter-strict-security**

## Command Default

Strict filter security is enabled.

## Modes

Interface configuration mode

## Usage Guidelines

When strict security mode is enabled, authentication for a port fails if the Filter-Id attribute contains invalid information, or if insufficient system resources are available to implement the IP ACLs.

When strict security mode is enabled:

- If the Filter-Id attribute in the Access-Accept message contains a value that does not refer to an existing filter (that is, IP ACL configured on the device), then the client will not be authorized, regardless of any other information in the message (for example, if the Tunnel-Private-Group-ID attribute specifies a VLAN on which to assign the port).
- If the device does not have the system resources available to dynamically apply a filter to a port, then the client will not be authenticated.

When strict filter security is disabled:

- If the Filter-Id attribute in the Access-Accept message contains a value that does not refer to an existing filter (that is, a MAC address filter or IP ACL configured on the device), then the client remains authorized and no filter is dynamically applied to it.
- By default, strict security mode is enabled for all MAC authentication and 802.1X-enabled interfaces, but you can manually disable or enable it using the **filter-strict-security** command from the authentication configuration mode or using the **authentication filter-strict-security** command from the interface configuration mode.

The **no** form of the command disables strict filter security.

## Examples

The following example enables strict filter security.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication filter-strict-security
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30mb | This command was modified. |

# authentication max-sessions

Specifies the maximum number of MAC sessions that can be authenticated per device or per port for MAC authentication and 802.1X authentication.

## Syntax

**authentication max-sessions** *count*

**no authentication max-sessions** *count*

## Command Default

The default number of MAC sessions that can be authenticated on a single interface is 2.

## Parameters

*count*

Specifies the maximum number of authenticated MAC sessions.

## Modes

Interface configuration mode

## Usage Guidelines

The maximum number of authenticated MAC sessions on an interface depends on the ICX device and dynamic ACL assignments.

If RADIUS assigns dynamic ACL to at least one client on the interface, the maximum number of MAC sessions that can be authenticated is limited to 32 in all ICX devices.

If dynamic ACL is not assigned to any of the clients on the interface, the maximum number of MAC addresses that can be authenticated varies depending on the ICX device as specified in Table 5.

System reload is not required for the changes to take effect. However, existing sessions are cleared for the changes to take effect.

**TABLE 5 Maximum number of authenticated MAC sessions per port on various platforms**

| Supported platforms | Maximum number of MAC sessions per port when none of the Clients has dynamic ACL | Maximum number of MAC sessions per port when at least one User has Dynamic ACL |
|---|---|---|
| ICX 7750 | 1024 | 32 |
| ICX 7650 | 1024 | 32 |
| ICX 7450 | 1024 | 32 |
| ICX 7250 | 1024 | 32 |

The system limit for authenticated MAC sessions also varies and depends on the ICX device and dynamic ACL assignments.

**TABLE 6 Maximum number of authenticated MAC sessions per system (standalone or stack) on various platforms**

| Supported platforms | Maximum number of MAC sessions per system when none of the clients has dynamic ACL | Maximum number of MAC sessions per system when at least one client has dynamic ACL |
|---|---|---|
| ICX 7750 | 1536 | 512 |
| ICX 7650 | 1536 | 512 |
| ICX 7450 | 1536 | 512 |
| ICX 7250 | 1536 | 512 |

The **no** form of the command reinstates the maximum authenticated MAC sessions allowed per port to the default value of 2.

# Examples

The following example specifies the maximum number of authenticated MAC sessions for port 1/1/1.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication max-sessions 30
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30b | The command was made available on ICX 7250, ICX 7450, and ICX 7750. The maximum number of authenticated MAC sessions per port was increased from 32 to 256 and 1024, depending on the platforms. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# authentication reauth-timeout

Sets the time to wait before reauthenticating a client after a timeout-action (success or critical-vlan) is applied. This command is applicable for MAC authentication and 802.1X authentication.

## Syntax

**authentication reauth-timeout** *seconds*

**no authentication reauth-timeout** *seconds*

## Command Default

The default re-authentication timeout is 300 seconds.

## Parameters

*seconds*

Sets the re-authentication timeout, in seconds. The range is from 60 to 4294967295.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form disables re-authentication timeout.

This command sets the re-authentication timeout at the interface level after the timeout action is specified as critical VLAN.

## Examples

The example shows specifying a re-authentication timeout of 120 seconds.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/2
device(config-if-e1000-1/1/2)# authentication reauth-timeout 120
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.61 | This command was modified to increase the default timeout from 60 to 300. |

# authentication source-guard-protection enable

Enables Source Guard Protection along with authentication on a specified interface.

## Syntax

**authentication source-guard-protection enable**

**no authentication source-guard-protection enable**

## Command Default

Source Guard Protection is not enabled.

## Modes

Interface configuration mode

## Usage Guidelines

When a new Flexible authentication session begins on a port that has Source Guard Protection enabled, the session either applies a dynamically created Source Guard ACL entry or it uses the dynamic IP ACL assigned by the RADIUS server. If a dynamic IP ACL is not assigned, the session uses the Source Guard ACL entry. The Source Guard ACL entry is **permit ip** *secure-ip* **any**, where *secure-ip* is obtained from the ARP Inspection table or from the DHCP Secure table. The DHCP Secure table is comprised of DHCP Snooping and Static ARP Inspection entries. The Source Guard ACL permit entry is added to the hardware table after all of the following events occur:

- The MAC address is authenticated
- The IP address is learned
- The MAC-to-IP mapping is checked against the Static ARP Inspection table or the DHCP Secure table

**NOTE**
In Flexible authentication, IP Source guard is applicable only for IPv4 traffic.

The Source Guard ACL entry is not written to the running configuration file. However, you can view the configuration using the **show mac-authentication sessions** command or the **show dot1x sessions** command at the global level or for a specific interface.

**NOTE**
The secure MAC-to-IP mapping is assigned at the time of authentication and remains in effect as long as the session is active. The existing session doesn't get affected if the DHCP Secure table is updated after the session is authenticated and while the session is still active.

The Source Guard ACL permit entry is removed when the session expires or is cleared.

The **no** form of the command disables source guard protection.

# Examples

The following example enables source guard protection on an interface.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication source-guard-protection enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.40a | IP Source guard was supported for 802.1X authentication-enabled port. |

# authentication timeout-action

Configures the authentication timeout actions to specify the action for the RADIUS server if an authentication timeout occurs.

## Syntax

**authentication timeout-action** { **success** | **failure** | **critical-vlan** }

**no authentication timeout-action** { **success** | **failure** | **critical-vlan** *vlan-id* }

## Command Default

The default authentication timeout action is failure.

## Parameters

**success**

Considers the client as authenticated after RADIUS timeout. After the timeout action is enabled as success, use the **no** form of the command to set the RADIUS timeout behavior to retry.

**failure**

Specifies the RADIUS timeout action to carry out the configured failure action. If the failure action is not configured, the client's MAC address is blocked in the hardware. Once the failure timeout action is enabled, use the **no** form of the command to reset the RADIUS timeout behavior to retry.

**critical-vlan**

On initial authentication, specifies that the client be moved to the client to the designated critical VLAN after authentication timeout. This command applies only to data traffic.

*vlan-id*

Specifies the ID of the VLAN to be configured as critical VLAN.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command will disable this functionality.

If the timeout is configured as success, client will be authenticated in the auth-default VLAN.

If the authentication failure action is configured as restricted VLAN using the **authentication fail-action** command, the client is placed in the restricted VLAN. A restricted VLAN must be configured using the **restricted-vlan** command at the global level or using the **authentication fail-action restricted-vlan** command at the interface level.

The critical VLAN specified at the interface level overrides the critical VLAN configured using the **critical-vlan** command at the global level. The configured critical VLAN configured at the global level will still be applicable to other ports that don't have critical VLAN configured at the interface level.

# Examples

The following example sets the **authentication timeout-action** command to success.

```
device(config)# authentication
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# authentication timeout-action success
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# authentication voice-vlan

Creates a voice VLAN ID for a port or for a group of ports.

## Syntax

**authentication voice-vlan** *num*

**no authentication voice-vlan** *num*

## Command Default

A local or port-level voice VLAN ID is not configured.

## Parameters

*num*

Specifies a valid VLAN ID. Valid values range from 1 through 4095.

## Modes

Interface configuration mode

## Usage Guidelines

When a local voice VLAN is configured, it facilitates continuing operation of IP phones in the following scenarios:

- The authentication process does not return a client VLAN.
- The RADIUS server is not reachable.
- Phone authentication fails.

The local voice VLAN configuration overrides the global voice VLAN configuration.

When a local voice VLAN is not configured or the local voice VLAN configuration is removed, the global VLAN configuration takes effect.

When you configure a voice VLAN ID on the port to which the VoIP phone is connected, the device automatically detects and reconfigures the VoIP phone when it is physically moved from one port to another within the same device.

When the ICX device receives the VoIP phone query, it sends the voice VLAN ID in a reply packet to the VoIP phone. The VoIP phone then configures itself within the voice VLAN.

As long as the port to which the VoIP phone is connected has a voice VLAN ID, the phone will configure itself into that voice VLAN. If you change the voice VLAN ID, the software will immediately send the new ID to the VoIP phone, and the VoIP phone will reconfigure itself with the new voice VLAN.

Some VoIP phones may require a reboot after configuring or reconfiguring a voice VLAN ID.

The **no** form of the command removes the voice VLAN ID from the port.

# Examples

The following example creates a VLAN ID for a port.

```
device(config)# interface ethernet 2/1/1
device(config-if-e1000-2/1/1)# authentication voice-vlan 1001
```

The following example creates a VLAN ID for a group of ports.

```
device(config)# interface ethernet 1/1/2 to 1/1/10
device(config-if-e1000-1/1/2-1/1/10)# authentication voice-vlan 1005
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# authentication-algorithm

Specifies the cryptographic algorithm to be used for the key in the keychain.

## Syntax

**authentication-algorithm** { **hmac-sha-1** | **hmac-sha-256** | **md5** | **sha-1** | **sha-256** }

**no authentication-algorithm** { **hmac-sha-1** | **hmac-sha-256** | **md5** | **sha-1** | **sha-256** }

## Command Default

An authentication algorithm is not specified by default.

## Parameters

**hmac-sha-1**
Sets the authentication algorithm to HMAC-SHA-1.

**hmac-sha-256**
Sets the authentication algorithm to HMAC-SHA-256.

**md5**
Sets the authentication algorithm to MD5.

**sha-1**
Sets the authentication algorithm to SHA-1.

**sha-256**
Sets the authentication algorithm to SHA-256.

## Modes

Key ID configuration mode

## Usage Guidelines

The application or protocol chooses the cryptographic algorithm that matches its criteria.

A key is considered valid only if the key has not expired, and the password and authentication algorithm have been specified.

The **no** form of the command removes the authentication algorithm from the key.

## Examples

The following example specifies MD5 to be used as the authentication algorithm.

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)# key-id 10
device(config-keychain-xprotocol-key-10)# authentication-algorithm md5
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# authentication-key

Defines an authentication key for Network Time Protocol (NTP).

## Syntax

**authentication-key key-id** *key-id* { **md5** | **sha1** } *key-string*

**no authentication-key key-id** *key-id* [ **md5** | **sha1** ] *key-string*

## Command Default

Authentication keys are not configured.

## Parameters

**key-id** *key-id*

Specifies a valid key ID. The value can range from 1 through 65535.

**md5**

Message authentication support is provided using the Message Digest 5 algorithm.

**sha1**

The SHA1 keyed hash algorithm is used for NTP authentication.

*key-string*

The value of the MD5 or SHA1 key. The length of the key string may be up to 16 characters. Up to 32 keys may be defined.

## Modes

NTP configuration mode

## Usage Guidelines

If Joint Interoperability Test Command (JITC) is enabled, only the **sha1** option is available.

If the NTP server or peer is configured without authentication keys, the NTP request is not sent to the configured server or peer.

The same set or subset of key ID and key string should be installed on all NTP devices.

The **no** form of the command removes the authentication key.

## Examples

The following example shows how to configure an authentication key.

```
device(config)# ntp
device(config-ntp)# authentication-key key-id 1 md5 moof
```

# auto-cost reference-bandwidth (OSPFv2)

Configures reference bandwidth.

## Syntax

**auto-cost reference-bandwidth** { *value* | **use-active-ports** }

**no auto-cost reference-bandwidth**

## Command Default

Reference bandwidth is 100 Mbps.

## Parameters

*value*

Reference bandwidth in Mbps. Valid values range from 1 through 4294967.

**use-active-ports**

Specifies that any dynamic change in bandwidth immediately affects the cost of OSPF routes. This parameter enables cost calculation for currently active ports only.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use this command to configure the cost of an interface that a device advertises to it's OSPF neighbors. OSPF calculates the cost of a route as the ratio of the reference bandwidth to the bandwidth of the egress interface. An increase in the reference bandwidth results in an increased cost. If the resulting cost is less than 1, the software rounds the cost up to 1.

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- LAG group — The combined bandwidth of all the ports.
- Virtual interface — The lowest individual bandwidth of all the ports that carry the VLAN for the associated VE.

If a change to the reference bandwidth results in a cost change to an interface, the device sends a link-state update to update the costs of interfaces advertised by the device.

**NOTE**
If you specify the cost for an individual interface (by using the **ip ospf cost** command), the cost you specify overrides the cost calculated by the software.

The **no** form of the command disables bandwidth configuration.

# Examples

The following example configures a reference bandwidth of 500.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port's cost = 500/10 = 50.
- 100 Mbps port's cost = 500/100 = 5.
- 1000 Mbps port's cost = 500/1000 = 0.5, which is rounded up to 1.

The costs for 10 Mbps and 100 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

# auto-cost reference-bandwidth (OSPFv3)

Configures reference bandwidth.

## Syntax

**auto-cost reference-bandwidth** *value*

**no auto-cost reference-bandwidth**

## Command Default

Reference bandwidth is 100 Mbps.

## Parameters

*value*

Reference bandwidth in Mbps. Valid values range from 1 through 4294967. The default is 100 Mbps.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command to configure the cost of an interface that a device advertises to it's OSPF neighbors. OSPFv3 calculates the cost of a route as the ratio of the reference bandwidth to the bandwidth of the egress interface. An increase in the reference bandwidth results in an increased cost. If the resulting cost is less than 1, the software rounds the cost up to 1.

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- LAG group — The combined bandwidth of all the ports.
- Virtual (Ethernet) interface — The combined bandwidth of all the ports in the port-based VLAN that contains the virtual interface.

If a change to the reference bandwidth results in a cost change to an interface, the device sends a link-state update to update the costs of interfaces advertised by the device.

**NOTE**
If you specify the cost for an individual interface using the **ipv6 ospf cost** command, the cost you specify overrides the cost calculated by the software.

Some interface types are not affected by the reference bandwidth and always have the same cost regardless of the reference bandwidth in use:

- The cost of a loopback interface is always 1.
- The cost of a virtual link is calculated using the Shortest Path First (SPF) algorithm and is not affected by the auto-cost feature.

- The bandwidth for tunnel interfaces is 9 Kbps and is subject to the auto-cost feature.

The **no** form of the command restores the reference bandwidth to its default value and, thus, restores the default costs of the interfaces to their default values.

# Examples

The following example configures a reference bandwidth of 500.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port's cost = 500/10 = 50.
- 100 Mbps port's cost = 500/100 = 5.
- 1000 Mbps port's cost = 500/1000 = 0.5, which is rounded up to 1.
- 155 Mbps port cost = 500/155 = 3.23, which is rounded up to 4
- 622 Mbps port cost = 500/622 = 0.80, which is rounded up to 1
- 2488 Mbps port cost = 500/2488 = 0.20, which is rounded up to 1

The costs for 10 Mbps, 100 Mbps, and 155 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

# auto-lacp

Configures the auto-LACP (Link Aggregation Control Protocol) deployment for a specific port or a range of ports.

## Syntax

**auto-lacp ethernet** *stack-id/slot/port* [ **ethernet** *stack-id/slot/port* | **to** *stack-id/slot/port* ]

**no auto-lacp ethernet** *stack-id/slot/port* [ **ethernet** *stack-id/slot/port* | **to** *stack-id/slot/port* ]

## Command Default

Auto-LACP is not deployed on any ports in the system.

## Parameters

**ethernet** *stack-id/slot/port*

Specifies the Ethernet port or the beginning range of the port list in terms of stack ID, slot number, or port number.

**to** *stack-id/slot/port*

Specifies the end range of the port list in terms of stack ID, slot number, or port number.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the auto-LACP configuration for a specific port or range of ports.

## Examples

The following example configures the auto-LACP deployment on the ports 1/1/7 and 2/1/11 to 2/1/12.

```
device(config)# auto-lacp ethernet 1/1/7 ethernet 2/1/11 to 2/1/12
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# autosave

Automatically saves learned secure MAC addresses to the startup configuration at specified intervals.

## Syntax

**autosave** *time*

**no autosave** *time*

## Command Default

By default, secure MAC addresses are not autosaved to the startup-config file.

## Parameters

*time*

The interval between two autosaves, in minutes. The valid range is from 15 to1440 minutes.

## Modes

Port security configuration mode

Port security interface configuration mode

## Usage Guidelines

The autosave feature saves learned MAC addresses by copying the running configuration to the startup configuration.

If you change the autosave interval, the next save occurs according to the old interval, and then the new interval takes effect. To change the interval immediately, disable autosave by entering the **no autosave** command, and then configure the new autosave interval using the **autosave** command.

The **no** form of the command disables autosave.

## Examples

The following example saves learned secure MAC addresses every 20 minutes automatically.

```
device(config)# port security
device(config-port-security)# autosave 20
```

The following example saves learned secure MAC addresses every 20 minutes automatically on an interface.

```
device(config)# port security
device(config-port-security)# interface ethernet 1/1/1
device(config-port-security-e1000-1/1/1)# autosave 20
```

# backup

Designates a virtual router as a Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRP-E) backup device and configures priority and track values.

## Syntax

**backup** [ **priority** *value* ] [ **track-priority** *value* ]

**no backup** [ **priority** *value* ] [ **track-priority** *value* ]

## Command Default

No virtual routers are designated as a VRRP or VRRP-E backup device.

## Parameters

**priority** *value*

Sets a priority value for a backup device. Values are from 8 through 254. In VRRP, the default backup device priority is 100, and the owner device has a default priority of 255. In VRRP-E, the default backup device priority is 100.

**track-priority** *value*

Sets the new priority value if the interface goes down. Values are from 1 through 254. Default is 2 for VRRP, and default is 5 for VRRP-E.

## Modes

VRID interface configuration mode

## Usage Guidelines

In VRRP, the backup device with the highest priority assumes the role of VRRP master device if the owner device fails. The interface on which the Virtual Routing ID (VRID) is configured must be in the same subnet (but not be the same address) as the IP address associated with the VRID by the owner device.

In VRRP-E, all devices are configured as backup devices and the backup device with the highest priority becomes the master device. If the master device fails, the backup device with the highest priority at that time assumes the role of VRRP master device. The IP address assigned to the interface of any device in the same virtual router must be in the same IP subnet. The IP address assigned to the VRID must not be configured on any of the ICX devices.

This command must be entered before the **ip-address** command can be configured for a VRRP or VRRP-E virtual routing ID.

The **no** form of this command removes the virtual router configuration.

# Examples

The following example configures the device as a VRRP backup device and assigns it a priority of 110.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.5.3/24
device(config-if-e1000-1/1/5)# ip vrrp vrid 1
device(config-if-e1000-1/1/5-vrid-1)# backup priority 110
device(config-if-e1000-1/1/5-vrid-1)# advertise backup
device(config-if-e1000-1/1/5-vrid-1)# ip-address 10.53.5.254
device(config-if-e1000-1/1/5-vrid-1)# activate
```

The following example configures the device as a VRRP-E backup device and assigns it a priority of 50 and a track priority of 10.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.10.4/24
device(config-if-e1000-1/1/5)# ip vrrp vrid 2
device(config-if-e1000-1/1/5-vrid-2)# backup priority 50 track-priority 10
device(config-if-e1000-1/1/5-vrid-2)# ip-address 10.53.10.254
device(config-if-e1000-1/1/5-vrid-2)# activate
```

# backup (VSRP)

Configures the device as a VSRP backup for the VRID or changes the backup priority and the track priority.

## Syntax

**backup** [ **priority** *priority-number* [ **track-priority** *track-number* ] ]

**no backup** [ **priority** *priority-number* [ **track-priority** *track-number* ] ]

## Command Default

The default backup priority for the VSRP VRID is 100.

The default track priority for all track ports is 5.

## Parameters

**priority** *priority-number*
Configures the backup priority for the VSRP VRID. The range is from 6 through 255. The default value is 100.

**track-priority** *track-number*
Configures the track priority for the VSRP VRID. The range is from 1 through 254. The default value is 5.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

This configuration is important because in VSRP, all devices on which a VRID are configured are backups. The master is then elected based on the VSRP priority of each device. There is no "owner" device as there is in VRRP.

The backup priority is used for election of the master. The VSRP backup with the highest priority value for the VRID is elected as the master for that VRID. If two or more backups are tied with the highest priority, the backup with the highest IP address becomes the master for the VRID.

The track priority is used with the track port feature. When you configure a VRID to track the link state of other interfaces, if one of the tracked interface goes down, the software changes the VSRP priority of the VRID interface. The software reduces the VRID priority by the amount of the priority of the tracked interface that went down. For example, if the VSRP interface priority is 100 and a tracked interface with track priority 60 goes down, the software changes the VSRP interface priority to 40. If another tracked interface goes down, the software reduces the VRID priority again, by the amount of the tracked interface track priority.

The **no** form of the command without any options removes the device as the backup. The **no** form of the command with the options resets the backup priority value and the track priority value to the default values.

# Examples

The following example configures the backup priority as 75.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup priority 75
device(config-vlan-200-vrid-1)# activate
```

The following example configures the backup priority as 100 and the track priority as 2.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup priority 100 track-priority 2
device(config-vlan-200-vrid-1)# activate
```

# backup-hello-interval

Configures the interval at which backup Virtual Router Redundancy Protocol (VRRP) routers advertise their existence to the master router.

## Syntax

**backup-hello-interval** *seconds*

**no backup-hello-interval** *seconds*

## Command Default

The default backup hello interval is 60 seconds.

## Parameters

*seconds*

The interval, in seconds, at which a backup VRRP router advertises its existence to the master router. Valid values range from 60 through 3600.

## Modes

VRID interface configuration mode

## Usage Guidelines

The interval is the length of time, in seconds, between each advertisement sent from the backup routers to the master router. The advertisement notifies the master router that the backup is still active. If the master router does not receive an advertisement from the backup router within a designated amount of time, the backup router with the highest priority can assume the role of master.

The **backup-hello-interval** command is configured only on VRRP backup routers and is supported by VRRP and VRRP Extended (VRRP-E).

The **no** form disables the advertisement of a VRRP backup router to a VRRP master router.

## Examples

The following example enables advertisements from the VRRP backup router and sets the hello message interval to 80 seconds.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# backup priority 90
device(config-if-e1000-1/1/6-vrid-1)# advertise backup
device(config-if-e1000-1/1/6-vrid-1)# backup-hello-interval 80
```

# backup-hello-interval (VSRP)

Configures the time interval during which Hello messages are sent by the backup.

## Syntax

**backup-hello-interval** *number*

**no backup-hello-interval** *number*

## Command Default

The backup sends a Hello message to the master every 60 seconds by default.

## Parameters

*number*

Specifies the time interval for the backup to send the Hello messages. The time range is from 60 through 3600 seconds.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The **no** form of the command resets the time interval to the default value.

## Examples

The following example changes the Hello message time interval.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup
device(config-vlan-200-vrid-1)# activate
device(config-vlan-200-vrid-1)# backup-hello-interval 180
```

# bandwidth (interface)

Sets and communicates bandwidth value for an interface to higher-level protocols such as OSPFv2 and OSPFv3, so this setting can be used to influence the routing cost for routes learnt on these interfaces.

## Syntax

**bandwidth** { *kilobits* }

**no bandwidth** { *kilobits* }

## Command Default

For physical ports, the port speed is the default bandwidth. For VE interfaces and Link aggregation (LAG) groups, the sum of port speeds of individual physical ports is the default bandwidth.

## Parameters

*kilobits*

Intended bandwidth, in kilobits per second. There is no default value for this parameter. The range is from 1 to 1000000000 kbps (100 Gbps).

## Modes

Interface configuration mode.

## Usage Guidelines

This command is supported on all Ruckus FastIron platforms.

You cannot adjust the actual bandwidth of an interface with this command. When you configure the interface bandwidth for virtual Ethernet that is associated with multiple physical interfaces, OSPF does not adjust its metric cost if one of those associated interfaces is down, and does not generate network and router link state advertisement.

he **no** form of the command removes the bandwidth value.

## Examples

The following example sets the bandwidth to 2000 kbps on a specific Ethernet interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1) bandwidth 2000
```

The following example sets the bandwidth to 2000 kbps on a specific virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# vlan 10
device(config-vlan-10)# interface ve 10
device(config-vif-10) bandwidth 2000
```

The following example sets the bandwidth to 2000 kbps on a specific tunnel interface.

```
device# configure terminal
device(config)# interface tunnel 2
device(config-tnif-2) bandwidth 2000
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was introduced. |

# banner

Defines a login banner.

## Syntax

**banner** [ **exec** | **incoming** ] *banner-string*

**no banner** [ **exec** | **incoming** ] *banner-string*

**banner motd** { *banner-string* | **require-enter-key** }

**no banner motd** { *banner-string* | **require-enter-key** }

## Command Default

A banner is not configured.

## Parameters

**exec**
Sets the EXEC process creation banner; that is, the message to be displayed when you enter the Privileged EXEC mode.

**incoming**
Sets the incoming terminal line banner; that is, the message to be displayed on the console when a user establishes a Telnet session.

*banner-string*
The ASCII string indicating the banner string in the format "c banner text c" where "c" is the delimiting character.

**motd**
Sets the message of the day (MOTD) banner; that is, the message to be displayed on a user terminal when a Telnet CLI session is established.

**require-enter-key**
Requires pressing of the Enter key after the MOTD message is displayed. This requirement is disabled by default. Unless configured, you do not have to press Enter after the MOTD banner is displayed.

## Modes

Global configuration mode

## Usage Guidelines

The *banner-string* includes a delimiting character. You begin and end the message with this delimiting character. The delimiting character can be any character except a double-quotation mark (") and cannot appear in the banner text. The banner text can be up to 4000 characters long, and can consist of multiple lines.

The **no** form of the command removes the banner. Use the **no banner motd require-enter-key** command to remove the requirement of pressing the Enter key once the banner text is displayed.

# Examples

The following example shows how to set a banner with "c" as the delimiting character.

```
device(config)# banner c Good Morning! c
```

The following example shows how to set a MOTD banner with "$" as the delimiting character.

```
device(config)# banner motd $ Welcome!!! $
```

The following example shows how to configure the requirement to press the Enter key after the banner message is displayed.

```
device(config)# banner motd require-enter-key
```

The following example shows the message displayed when the requirement to press the Enter key is enabled upon accessing the switch from Telnet.

```
Authorized Access Only ...
Press <Enter> to accept and continue the login process....
```

# batch buffer

Creates a group of CLI commands per batch ID that is used in the automatic execution of commands in batches.

## Syntax

**batch buffer** *batch-id delimiting-character command-list delimiting-character*

**no batch buffer** *batch-id*

## Command Default

CLI commands are not grouped per batch.

## Parameters

*batch-id*

Specifies the unique batch buffer ID. The value range is from 1 through 4.

*delimiting-character*

Enables an onboard editor on which the list of CLI commands is added. The second occurrence of the delimiting character closes the onboard editor.

*command-list*

Specifies the list of commands that you want to add in the batch buffer. A maximum of 10 commands can be added in a batch buffer.

## Modes

Global configuration mode

## Usage Guidelines

You can create only up to 4 batches and each batch can have a maximum of 10 commands.

The commands that are present at the user EXEC mode, privileged EXEC mode, global configuration mode, and sub-level commands can be added to a batch.

The commands that are saved in the batch buffer are applied on the device only if the **execute batch** command is issued.

The following list of commands cannot be issued using the batch process:

- At the privileged EXEC level:
  - **exit**
  - **ping**
  - **reload**
  - **telnet**
  - **quit**
  - **traceroute**

  - **ssh**
- At the global configuration level:
  - **quit**
  - **relative-utilization**
  - **batch**

Any command that requires user intervention will fail during batch execution.

The **no** form of the command removes the configured batch.

# Examples

The following example creates a batch buffer containing two CLI commands.

```
device# configure terminal
device(config)# batch buffer 1 &
configure terminal
hostname ruckus &
```

# bgp-redistribute-internal

Causes the device to allow the redistribution of IBGP routes from BGP into OSPF for non-default VRF instances.

## Syntax

**bgp-redistribute-internal**

**no bgp-redistribute-internal**

## Command Default

This feature is disabled.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

This example enables BGP4 route redistribution.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# bgp-redistribute-internal
```

This example enables BGP4+ route redistribution in BGP address-family IPv6 unicast configuration mode.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# bgp-redistribute-internal
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# block

Configures the time users must wait before the next cycle of Web Authentication begins after they have exceeded the limit for Web Authentication attempts.

## Syntax

**block** [ **mac** *mac-address* ] **duration** *time*

**no block** [ **mac** *mac-address* ] **duration** *time*

## Command Default

The default is 90 seconds.

## Parameters

**mac** *mac-address*

Configures the host with the specified MAC address to be temporarily or permanently blocked from attempting Web Authentication.

**duration** *time*

Configures the time duration users must wait before the next cycle of Web Authentication attempts is allowed. Valid values are from 0 through 128,000 seconds. The default is 90 seconds, and entering 0 means the user is infinitely blocked.

## Modes

Web Authentication configuration mode

## Usage Guidelines

To unblock the MAC address, wait until the block duration timer expires or enter the **clear webauth vlan** *vlan-id* **block-mac** command.

The **no** form of the command resets the duration time to the default.

## Examples

The following example configures the block duration to 1000 seconds.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# block duration 1000
```

The following example configures the block duration for a specific host.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# block mac 1111.2222.3333 duration 1000
```

# block-applicant

Disables the VLAN advertising on a GVRP-enabled port.

## Syntax

**block-applicant all**

**block-applicant ethernet** *unit/slot/port* [ [ **ethernet** *unit/slot/port* ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *unit/slot/port* ]... ]

**block-applicant lag** *lag-id* [ [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *lag-id* ]... ]

**no block-applicant all**

**no block-applicant ethernet** *unit/slot/port* [ [ **ethernet** *unit/slot/port* ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *unit/slot/port* ]... ]

**no block-applicant lag** *lag-id* [ [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *lag-id* ]... ]

## Command Default

VLANs are advertised on GVRP-enabled ports.

## Parameters

**all**
Disables VLAN advertising on all GVRP-enabled ports.

**ethernet** *unit/slot/port*
Disables VLAN advertisement on the specified GVRP-enabled Ethernet port.

**to** *unit/slot/port*
Specifies the range of GVRP-enabled Ethernet ports on which you want to disable VLAN advertising.

**lag** *lag-id*
Specifies the LAG virtual interface.

## Modes

GVRP configuration mode

## Usage Guidelines

**NOTE**
Even when VLAN adverstising is disabled, Leaveall messages are still sent on the GVRP ports.

The **no** form of the command allows the VLAN advertising on GVRP-enabled ports.

# Examples

The following example shows how to disable VLAN advertising on all ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-applicant all
```

The following example shows how to disable VLAN advertising on specific ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-applicant ethernet 1/1/24 ethernet 1/6/24 ethernet 1/8/17
```

The following example shows how to disable VLAN advertising on a range of ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-applicant ethernet 1/1/1 to 1/1/8
```

The following example shows how to disable VLAN advertising on a list of specific ports as well as on a range of ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-applicant ethernet 1/1/1 to 1/1/8 ethernet 1/1/24 ethernet 1/6/24 ethernet
1/8/17
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* option. |

# block-learning

Disables the VLAN learning on GVRP-enabled ports.

## Syntax

**block-learning all**

**block-learning ethernet** *unit/slot/port* [ [ **ethernet** *unit/slot/port* ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *unit/slot/port* ]... ]

**block-learning lag** *lag-id* [ [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *lag-id* ]... ]

**no block-learning all**

**no block-learning ethernet** *unit/slot/port* [ [ **ethernet** *unit/slot/port* ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *unit/slot/port* ]... ]

**no block-learning lag** *lag-id* [ [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ] [ **lag** *lag-id* [ **to** *lag-id* ] ] [ **to** *lag-id* ]... ]

## Command Default

VLAN learning is enabled.

## Parameters

**all**
Disables VLAN learning on all GVRP-enabled ports.

**ethernet** *unit/slot/port*
Disables VLAN learning on the specified Ethernet interface.

**to** *unit/slot/port*
Specifies a range of GVRP-enabled Ethernet ports on which you want to disable VLAN learning.

**lag** *lag-id*
Specifies the LAG virtual interface.

## Modes

GVRP configuration mode

## Usage Guidelines

**NOTE**
The port still advertises VLAN information unless you also disable VLAN advertising.

The **no** form of the command re-enables VLAN learning.

# Examples

The following example shows how to disable VLAN learning on all GVRP-enabled ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-learning all
```

The following example shows how to disable VLAN learning on a list of specific GVRP-enabled ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-learning ethernet 1/1/24 ethernet 1/6/22 ethernet 1/8/17
```

The following example shows how to disable VLAN learning on a range of GVRP-enabled ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-learning ethernet 1/1/1 to 1/1/8
```

The following example shows how to disable VLAN learning on a list of ports along with a range of GVRP-enabled ports.

```
device(config)# gvrp-enable
device(config-gvrp)# block-learning ethernet 1/1/1 to 1/1/8 ethernet 1/1/24 ethernet 1/6/24 ethernet
1/8/17
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* option. |

# boot system flash

Configures the device to boot from the image stored in the flash memory.

## Syntax

**boot system flash** { **primary** | **secondary** } [ **yes** ]

**no boot system flash** { **primary** | **secondary** } [ **yes** ]

## Command Default

By default, the device first attempts to boot from the image stored in its primary flash, then its secondary flash, and then from a TFTP server.

## Parameters

**primary**

Configures to boot from the image stored in its primary flash.

**secondary**

Configures to boot from the image stored in its secondary flash.

**yes**

Confirms the system boot preference settings. This option is equivalent to using the **write memory** command. This option is available only in Privileged EXEC mode.

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

You can use boot commands to immediately initiate software boots from a software image stored in the primary or secondary flash on an ICX device.

It is very important that you verify a successful transfer of the boot code before you reset the system. If the boot code is not transferred successfully but you try to reset the system, the system will not have the boot code with which to successfully boot.

You can modify the default booting sequence in the global configuration mode using the **boot system** command.

Execute the **write memory** command to save the boot preferences to the startup configuration. If you are executing the **boot system flash** command from the Privileged EXEC mode, you can use the **yes** option to save the boot preference to the startup configuration. Executing the **write memory** command is not required in this case.

You can use the **show boot-preference** command to view the boot sequence preference.

The **no** form of the command resets the boot preference to the default.

# Examples

The following example shows how to set the system to boot the image from the secondary flash.

```
device(config)# boot system flash secondary
```

The following example shows how to set the system to boot the image from the primary flash and save the preference to the startup configuration.

```
device# boot system flash primary yes
```

# boot system tftp

Configures the device to boot from the image stored on a TFTP server.

## Syntax

**boot system tftp** *server-ip file-name* [ **fiber-port** ]

**no boot system tftp** *server-ip file-name* [ **fiber-port** ]

## Command Default

By default, the device first attempts to boot from the image stored in its primary flash, then its secondary flash, and then from a TFTP server.

## Parameters

*server-ip*

The IP address of the TFTP server. The IP address of the device and the TFTP server should be in the same subnet.

*file-name*

The boot code file name.

**fiber-port**

Configures to boot the device from a TFTP server through the fiber connection. This option is available only in devices running router images and in Privilege EXEC mode.

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

It is very important that you verify a successful transfer of the boot code before you reset the system. If the boot code is not transferred successfully but you try to reset the system, the system will not have the boot code with which to successfully boot.

The **boot system tftp** command is not supported in a stacking environment.

The **no** form of the command resets the boot preference to the default.

## Examples

The following example shows how to configure the device to boot from the image stored on a TFTP server.

```
device# boot system tftp 192.168.10.1 SPS08040.bin
```

# bootfile

Specifies the boot image to be used by the client.

## Syntax

**bootfile** *name*

## Parameters

*name*

Specifies the name of the bootfile to be used by the client.

## Modes

DHCP server pool configuration mode

## Examples

The following example specifies the bootfile name.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# bootfile foxhound
```

# bootp-relay-max-hops

Modifies the maximum number of BootP or DHCP hops.

## Syntax

**bootp-relay-max-hops** *max-hop*

**no bootp-relay-max-hops** *max-hop*

## Command Default

By default, a Ruckus Layer 3 switch forwards a BootP or DHCP request if its hop count is four or less, but discards the request if the hop count is greater than four.

## Parameters

*max-hop*

Specifies the maximum number of hops. The parameter value can be from 1 through 15.

## Modes

Global configuration mode

## Usage Guidelines

Because the hop count value initializes at zero, the hop count value of an ingressing DHCP Request packet is the number of Layer 3 routers that the packet has already traversed.

## Examples

The following example modifies the maximum number of BootP or DHCP hops to 10. The example allows the Layer 3 switch to forward BootP or DHCP requests that have passed through 10 previous hops before reaching the Layer 3 switch. Requests that have traversed 11 hops before reaching the switch are dropped.

```
device(config)# bootp-relay-max-hops 10
```

# bpdu-flood-enable

Configures the MCT cluster devices to flood the SSTP or MSTP BPDUs in the SSTPor MSTP domain.

## Syntax

**bpdu-flood-enable**

**no bpdu-flood-enable**

## Command Default

BPDU flooding is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

When **bpdu-flood-enable** is configured, there should not be any links other than the ICL connecting the two MCT cluster devices. If there is an additional link, then the flooded BPDU will cause a loop and high CPU utilization.

**NOTE**
The **bpdu-flood-enable** command is not supported on ICX 7750.

The **no** form of the command disables the BPDU flooding.

## Examples

The following example shows how to configure BPDU flooding on the device.

```
device(config)# bpdu-flood-enable
Warning - Any recieved untagged BPDUs will now be flooded to all the ports.
```

# breakout ethernet

Configures sub-ports from 40 Gbps ports.

## Syntax

**breakout ethernet** *unit*/*slot*/*port*

**breakout ethernet** *unit*/*slot*/*port* **to ethernet** *unit*/*slot*/*port*

**breakout ethernet** *unit*/*slot*/*port* **ethernet** *unit*/*slot*/*port*

**no breakout ethernet** *unit*/*slot*/*port*

**no breakout ethernet** *unit*/*slot*/*port* **to ethernet** *unit*/*slot*/*port*

**no breakout ethernet** *unit*/*slot*/*port* **ethernet** *unit*/*slot*/*port*

## Command Default

By default, ports that can be broken out are configured as 40 Gbps ports.

## Parameters

**ethernet**

Specifies the connection as ethernet.

*unit*/*slot*/*port*

Specifies the port to be broken into 10 Gbps sub-ports. If there are two port identifiers in the command line, the first port designates the beginning port in a range of ports to be broken out, and the second port indicates the end of the breakout range. When a range is specified, the 10 Gbps sub-ports within the range are implicitly included.

**to**

Designates a range of ports to be configured when followed by an ending port identifier. This is an optional keyword.

## Modes

Global configuration mode.

## Usage Guidelines

Use the **no** form of the command to remove breakout configuration from the designated port or range of ports.

No configuration may be present on a port for which the **breakout ethernet** command is issued. When the command is issued on a port with pre-existing configuration, an error message is returned. The existing configuration must be removed before the **breakout ethernet** command is re-issued.

The **breakout ethernet** command is available only on certain ICX 7750 40 Gbps ports. Refer to the *FastIron Ethernet Switch Administration Guide* for a table of available breakout ports. Refer to the *ICX 7750 Ethernet Switch Hardware Installation Guide* for detailed information on breakout cables.

The **breakout ethernet** command can be issued on stand-alone units only. Stacking cannot be enabled on a port configured for breakout. An error is returned if you try to enable stacking on a unit that has any breakout ports configured. The breakout configuration must be removed manually before stacking can be enabled. Use the **show breakout** command to display the breakout configuration for a unit.

The **breakout ethernet** and **no breakout ethernet** commands must be followed by a **write memory** command and a **reload** command for the port configuration changes to take effect.

# Examples

The following example configures breakout on port 1/1/5, after existing configuration on the port is removed.

```
Device# configure terminal
Device(config)# breakout ethernet 1/1/5
Error: Port 1/1/5 has sflow forwarding
Device(config)# interface ethernet 1/1/5
Device(config-if-e40000-1/1/5)# no sflow forwarding
Device(config-if-e40000-1/1/5)# end
Device# write memory
Write startup-config done.
Device# configure terminal
Device(config)# breakout ethernet 1/1/5
Reload required.  Please write memory and then reload or power cycle.
Device(config)# write memory
Write startup-config done.
Device(config)# Flash Memory Write (8192 bytes per dot) .
Copy Done.
Device(config)# end
Device# reload
```

The following example checks for ports with active breakout configuration and then removes breakout from ports 1/3/1 through 1/3/6.

```
Device# show breakout

Unit-Id: 1

Port       Module Exist    Module Conf     breakout_conf   breakout_oper
1/1/5      Yes             No              Yes             Yes
1/1/6      Yes             No              Yes             Yes
1/1/7      Yes             No              Yes             Yes
1/1/8      Yes             No              Yes             Yes
1/1/9      Yes             No              Yes             Yes
1/1/10     Yes             No              Yes             Yes
1/1/11     Yes             No              Yes             Yes
1/1/12     Yes             No              Yes             Yes
1/1/13     Yes             No              Yes             Yes
1/1/14     Yes             No              Yes             Yes
1/1/15     Yes             No              Yes             Yes
1/1/16     Yes             No              Yes             Yes
1/2/1      Yes             No              Yes             Yes
1/2/2      Yes             No              Yes             Yes
1/2/3      Yes             No              Yes             Yes
1/2/4      Yes             No              Yes             Yes
1/2/5      Yes             No              Yes             Yes
1/2/6      Yes             No              Yes             Yes
1/3/1      Yes             No              Yes             Yes
1/3/2      Yes             No              Yes             Yes
1/3/3      Yes             No              Yes             Yes
1/3/4      Yes             No              Yes             Yes
1/3/5      Yes             No              Yes             Yes
1/3/6      Yes             No              Yes             Yes

Device# configure terminal
Device(config)# no breakout ethernet 1/3/1 to 1/3/6
Reload required.  Please write memory and then reload or power cycle.
Device(config)# write memory
Write startup-config done.

Device(config)# Flash Memory Write (8192 bytes per dot) .
Copy Done.
Device(config)# end
Device# reload
```

**NOTE**

If there had been any configuration on any sub-ports (1/3/1:1 to 1/3/6:4), the **no breakout** command would have returned an error. The configuration would then have to be removed from the sub-ports before breakout configuration could be removed.

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# broadcast client

Configures a device to receive Network Time Protocol (NTP) broadcast messages on a specified interface.

## Syntax

**broadcast client**

**no broadcast client**

## Command Default

The broadcast mode is not enabled.

## Modes

NTP interface configuration mode

## Usage Guidelines

An NTP broadcast client can be enabled on a maximum of 16 Ethernet interfaces. If the interface is operationally down or if NTP is disabled, the NTP broadcast server packets are not received.

The **no** form of the command disables the capability of a device to receive NTP broadcast messages.

## Examples

The following example configures a device to receive NTP broadcast messages on a specified interface.

```
device(config)# ntp
device(config-ntp)# ntp-interface management 1
device(config-ntp-mgmt-1)# broadcast client
```

# broadcast destination

Configures Network Time Protocol (NTP) broadcast destination options.

## Syntax

**broadcast destination** *ip-address* [ **key** *key-id* ] [ **version** *version-number* ]

**no broadcast destination** *ip-address* [ **key** *key-id* ] [ **version** *version-number* ]

## Command Default

The broadcast mode is not enabled.

## Parameters

*ip-address*
Specifies the IPv4 subnet address of the device to send NTP broadcast messages.

**key** *key-id*
Specifies the authentication key ID. By default, no authentication key is configured. Valid values are from 1 through 65535.

**version** *version-number*
Specifies the NTP version number.The version options are 3 and 4. The default value is 4.

## Modes

NTP interface configuration mode

## Usage Guidelines

The NTP broadcast server can be enabled on a maximum 16 Ethernet interfaces and four subnet addresses per interface. If the interface is operationally down or there is no IP address configured for the subnet address, the NTP broadcast server packets are not sent.

**NOTE**
This command is not effective if the NTP server is disabled.

The **no** form of the command disables the broadcast option.

## Examples

The following example configures NTP broadcast destination options.

```
device(config)# ntp
device(config-ntp)# ntp-interface management 1
device(config-ntp-mgmt-1)# broadcast destination 10.20.99.0 key 2 version 3
```

# broadcast limit (enable)

Configures the maximum number of broadcast packets allowed per second.

## Syntax

**broadcast limit** *num* **kbps**

**no broadcast limit** *num* **kbps**

## Command Default

Broadcast rate limiting is disabled.

## Parameters

*num*

Specifies the maximum number of broadcast packets per second. The value can be 1 to 8388607.

**kbps**

Enables byte-based limiting. The value can be 1 to Max Port Speed.

## Modes

Interface configuration mode

## Usage Guidelines

Use 0 or the **no** form of the command to disable broadcast rate limiting.

## Examples

The following example enables a broadcast rate limit of 131072 kbps.

```
device(config)# interface ethernet 9/1/1
device(config-if-e1000-9/1/1)# broadcast limit 131072 kbps
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | The command was introduced. |

# broadcast limit (logging)

Enables Syslog logging of broadcast packets.

## Syntax

**broadcast limit** *num* **kbps** [ **log** ]

**no broadcast limit** *num* **kbps** [ **log** ]

## Command Default

Broadcast rate logging is disabled.

## Parameters

*num*

Specifies the maximum number of broadcast packets per second. The value can be 1 to 8388607.

**kbps**

Enables byte-based limiting. The value can be 1 to Max Port Speed.

**log**

Enables Syslog logging when the broadcast limit exceeds *num* **kpbs** .

## Modes

Interface configuration mode

## Usage Guidelines

Use 0 or the **no** form of the command to disable broadcast rate logging.

## Examples

The following example enables broadcast limit logging when the configured broadcast limit exceeds 100 Kbps.

```
device(config)# interface ethernet 1/2/1
device(config-if-e10000-1/2/1)# broadcast limit 100 kbps log
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | The command was introduced. |
| 8.0.40a | The command was modified to include the keyword **log**. |

# bsr-candidate

Configures a bootstrap router (BSR) as a candidate to distribute rendezvous point (RP) information to the other PIM Sparse devices within a PIM Sparse domain.

## Syntax

**bsr-candidate ethernet** *unit/slot/port hash-mask-length* [ *priority* ]

**bsr-candidate lag** *lag-id hash-mask-length* [ *priority* ]

**bsr-candidate loopback** *num hash-mask-length* [ *priority* ]

**bsr-candidate ve** *num hash-mask-length* [ *priority* ]

**bsr-candidate tunnel** *num hash-mask-length* [ *priority* ]

**no bsr-candidate**

## Command Default

The PIM router does not participate in BSR election.

## Parameters

**ethernet** *unit/slot/port*
Specifies the Ethernet interface for the candidate BSR.

**lag** *lag-id*
Specifies the LAG virtual interface.

**loopback** *num*
Specifies the loopback interface for the candidate BSR.

**ve** *num*
Specifies the virtual interface for the candidate BSR.

**tunnel** *num*
Specifies a GRE tunnel interface.

*hash-mask-length*
Specifies the number of bits in a group address that are significant when calculating the group-to-RP mapping. The range is 1 to 32.

**NOTE**
It is recommended that you specify 30 for IPv4 networks.

*priority*
Specifies the BSR priority. The range is from 0 to 255, from low to high. The default is 0.

## Modes

PIM Router configuration mode

# Usage Guidelines

The **no** form of this command makes the PIM router cease to act as a candidate BSR.

Each PIM Sparse domain has one active BSR. For redundancy, you can configure ports on multiple devices as candidate BSRs. The PIM Sparse protocol uses an election process to select one of the candidate BSRs as the BSR for the domain. The BSR with the highest BSR priority is elected. If the priorities result in a tie, the candidate BSR interface with the highest IP address is elected.

Although you can configure the device as only a candidate BSR or an RP, it is recommended that you configure the same interface on the same device as both a BSR and an RP.

# Examples

The following example uses a physical interface to configure a device as a candidate BSR.

```
device(config)# router pim
Device(config-pim-router)# bsr-candidate ethernet 1/2/2 30 255
```

The following example uses a loopback interface to configure a device as a candidate BSR.

```
device(config)# router pim
device(config-pim-router)# bsr-candidate loopback 1 30 240
```

The following example uses a virtual interface to configure a device as a candidate BSR.

```
device(config)# router pim
device(config-pim-router)# bsr-candidate ve 120 30 250
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was modified to add the **tunnel** keyword. |
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# bsr-msg-interval

Sets the PIM BSR message interval timer.

## Syntax

**bsr-msg-interval** *time*

**no bsr-msg-interval** *time*

## Command Default

The default IPv6 PIM BSR message interval timer is 60 seconds.

## Parameters

*time*

Defines the interval at which the BSR sends RP candidate data to all IPv6-enabled routers within the IPv6 PIM Sparse domain. Valid values are 10 to 65535 seconds. The default is 60 seconds.

## Modes

IPv6 router PIM configuration

## Usage Guidelines

The BSR message interval timer defines the interval at which the BSR sends RP candidate data to all IPv6-enabled routers within the IPv6 PIM Sparse domain

The **no** form of the command resets the IPv6 PIM BSR message interval timer to the default value of 60 seconds.

## Examples

The following example sets the IPv6 PIM BSR message interval timer to 16 seconds.

```
device(config)# ipv6 router pim
device(config-ipv6-pim-router)# bsr-msg-interval 16
```

The following example sets the IPv6 PIM BSR message interval timer to 16 seconds for a specified VRF.

```
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# bsr-msg-interval 16
```

# buffer-profile port-region

Configures a buffer profile on a device.

## Syntax

**buffer-profile port-region** *port-region* **qd-buffer-profile** *user-profile-name*

**no buffer-profile port-region** *port-region* **qd-buffer-profile** *user-profile-name*

**buffer-profile port-region** *port-region* **scheduler-profile** *user-profile-name*

**no buffer-profile port-region** *port-region* **scheduler-profile** *user-profile-name*

**buffer-profile port-region** *port-region* **voip downlink 100 uplink 1000**

**no buffer-profile port-region** *port-region* **voip downlink 100 uplink 1000**

## Command Default

Buffer profiles are not configured.

## Parameters

*port-region*
Specifies the device number on which the user-configurable buffer profile is applied. The port-region number can be 0 through 15.

**qd-buffer-profile** *user-profile-name*
Applies the user-defined buffer profile.

**scheduler-profile** *user-profile-name*
Configures a defined scheduler profile.

**voip**
Configures a VoIP buffer profile.

**downlink 1000**
Configures the downlink ports as 1000 Megabits.

**uplink 100**
Configures the uplink ports as 100 Megabits.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command deletes the buffer profile configuration.

# Examples

The following example applies the buffer profile named profile1 to a device.

```
device(config)# buffer-profile port-region 0 qd-buffer-profile profile1
```

# buffer-sharing-full

Removes the buffer allocation limits on all ports and all traffic classes globally.

## Syntax

**buffer-sharing-full**

## Modes

Global configuration mode

## Usage Guidelines

The **buffer-sharing-full** command sets the total transmit queue depth limit and the transmit queue depth limits for each traffic class to 4095 for all ports of the device. The command overrides any existing individually configured queue depth limits. The command permits all available buffers in a port region to be used on a first-come, first-served basis by any of its ports, regardless of priority.

**NOTE**
The **buffer-sharing-full** command should be used carefully. By entering this command, there is no limit on the number of buffers a port or a specific priority on a port can use. One port could potentially use up all the available buffers of its port region and cause starvation on other ports of the port region. The command can create unpredictable behavior during traffic congestion or a blocking scenario, compromising network stability (by losing control packets), QoS, and stacking.

## Examples

The following example removes the buffer allocation limits on all ports and all traffic classes globally.

```
device(config)# buffer-sharing-full
```

# Commands C

## capability as4

Enables 4-byte autonomous system number (ASN) capability at the BGP global level.

### Syntax

**capability as4** { **disable** | **enable** }

**no capability as4** { **disable** | **enable** }

### Command Default

This feature is disabled.

### Parameters

**disable**

Disables 4-byte ASN capability at the BGP global level.

**enable**

Enables 4-byte ASN capability at the BGP global level.

### Modes

BGP configuration mode

### Usage Guidelines

Use the **no** form of this command to disable this functionality.

### Examples

To enable 4-byte ASN capability:

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# capability as4 enable
```

# captive-portal

Creates a user-defined Captive Portal profile.

## Syntax

**captive-portal** *profile-name*

**no captive-portal** *profile-name*

## Parameters

*profile-name*

Specifies the name of the user-defined Captive Portal profile.

## Modes

Global configuration mode

## Usage Guidelines

The Captive Portal profile serves as a template that includes configuration details specific to the external web server, such as virtual IP address, HTTP or HTTPS protocol port number, and login-page details hosted on the external captive portal server.

The details configured in the Captive Portal profile enable the switch to handle HTTP redirection mechanism and redirects the client to the login page hosted on the external captive portal server.

The Captive Portal profile can be attached to an external Web Authentication-enabled VLAN using the **captive-portal profile** command.

The **no** form of the command removes the Captive Portal profile.

## Examples

The following example creates the user-defined Captive Portal profile cp_ruckus.

```
device(config)# captive-portal cp_ruckus
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j |

# captive-portal profile

Applies a configured Captive Portal profile on a Web Authentication-enabled VLAN.

## Syntax

**captive-portal profile** *profile-name*

**no captive-portal profile** *profile-name*

## Command Default

A Captive Portal profile is not applied on a Web Authentication-enabled VLAN.

## Parameters

*profile-name*

Specifies the Captive Portal profile to be applied on a Web Authentication-enabled VLAN.

## Modes

Web Authentication configuration mode

## Usage Guidelines

The **no** form of the command removes the Captive Portal profile from the Web Authentication-enabled VLAN.

## Examples

The following example binds the Captive Portal profile cp_ruckus on Web Authentication-enabled VLAN 10.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# captive-portal profile cp_ruckus
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j |

# cdp enable

Enables Cisco Discovery Protocol (CDP) at the interface level.

## Syntax

**cdp enable**

**no cdp enable**

## Command Default

CDP is not enabled. CDP is enabled on an interface once CDP is enabled on the device.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables CDP on an interface.

## Examples

The following example enables CDP on an interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# cdp enable
```

# cdp run

Enables the device to intercept and display Cisco Discovery Protocol (CDP) messages.

## Syntax

**cdp run**

**no cdp run**

## Command Default

CDP is disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

This command also enables the device to detect CDP power requirements.

The **no** form of the command disables the device from intercepting and displaying CDP messages.

## Examples

The following example enables the device to intercept and display CDP messages.

```
device(config)# cdp run
```

# chassis fanless

Enables chassis fanless mode that sets the device to operate with the fan disabled while providing a PoE budget of 150 watts.

## Syntax

**chassis fanless** { *unit-id* | **all** }

**no chassis fanless** { *unit-id* | **all** }

## Command Default

Fanless mode is disabled.

## Parameters

*unit-id*

Enables fanless mode for a specified unit.

*all*

Enables fanless mode in all supported units of stack.

## Modes

Global configuration mode

## Usage Guidelines

Fanless mode is supported only on ICX 7150-24P and ICX 7150-48P devices.

Fanless mode can be enabled only if the PoE power allocation is less than or equal to 150W. If the PoE power allocation is more than 150W, PoE load must be reduced by removing PoE interfaces manually or by unplugging PoE devices.

When fanless mode is enabled, the fan speed is set to zero RPM.

Fanless mode is enabled from the active console.

Even if fanless mode is configured on a switch, fans will be turned on temporarily during boot up or reboot and will be turned off after the boot up.

The **no** form of the command resets the fan speed to auto and reinstates the PoE budget to the default value.

## Examples

The following example enables fanless mode on the device.

```
device(config)# chassis fanless 1
```

The following example enables fanless mode on all supported units of stack.

```
device(config)# chassis fanless all
```

# History

| Release version | Command history |
|---|---|
| 08.0.60 | This command was introduced. |
| 08.0.61 | This command was enhanced to support stacking. Also, the command name was modified from **chassis fanless-mode-enable** to **chassis fanless** with the **all** option. |

# chassis name

Configures a chassis name.

## Syntax

**chassis name** *name*

**no chassis name** *name*

## Command Default

A chassis name is not configured.

## Parameters

*name*

Specifies the name of the chassis.

## Modes

Global configuration mode

## Usage Guidelines

This command does not change the CLI prompt. Instead, the command assigns an administrative ID to the device.

The **no** form of the command removes the chassis name.

## Examples

The following example configures a chassis name.

```
device(config)# chassis name ch_2
```

# clear access-list

Clears ACL counters.

## Syntax

**clear access-list** { **all** | *std-acl-num* | *ext-acl-num* }

## Parameters

**all**

Clears all ACL counters.

*std-acl-num*

Clears the counter for the specified standard ACL. Valid values are from 1 through 99.

*extd-acl-num*

Clears the counter for the specified extended ACL. Valid values are from 100 through 199.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example clears all the ACL counters.

```
device# clear access-list all
```

The following example clears the counter for the standard ACL 10.

```
device# clear access-list 10
```

# clear access-list accounting

Clears access control list (ACL) accounting statistics for IPv4 ACLs, IPv6 ACLs, and Layer 2 MAC filters.

## Syntax

**clear access-list accounting all**

**clear access-list accounting** *interface-type interface-name* [ **in** | **out** ] [ **IPv4** | **IPv6** ]

**clear access-list accounting traffic-policy** { **all** | *name* }

## Parameters

**all**
Clears all statistics for all ACLs.

*interface-type interface-name*
Specifies the interface type (Ethernet, virtual interface, or LAG) and the ID of the interface.

**in**
Clears statistics of the inbound ACLs. If no direction is set, statistics for both inbound and outbound are cleared.

**out**
Clears statistics of the outbound ACLs. If no direction is set, statistics for both inbound and outbound are cleared.

**IPv4**
Clears statistics for IPv4 ACLs. Statistics for both IPv4 and IPv6 ACLs are cleared if this value is not set.

**IPv6**
Displays statistics for IPv6 ACLs. Statistics for both IPv4 and IPv6 ACLs are cleared if this value is not set.

**traffic-policy**
Clears traffic-policy statistics.

**all**
Clears all traffic-policy statistics.

*name*
Clears statistics of a specific traffic policy.

## Modes

Privileged EXEC mode

## Examples

The following example clears ACL accounting statistics for all configured ACLs.

```
device# clear access-list accounting all
```

The following example clears ACL accounting statistics, for inbound ACLs, for a specific port.

```
device# clear access-list accounting ethernet 1/1/5 in
```

The following example clears all traffic-policy statistics.

```
device# clear access-list accounting traffic-policy all
```

The following example clears ACL accounting statistics, for outbound IPv6 ACLs, for a specific port.

```
device# clear access-list accounting ve 100 out ipv6
```

The following example clears ACL accounting statistics, for outbound IPv4 ACLs, for a specific port.

```
device# clear access-list accounting ve 100 out ipv4
```

The following example clears ACL accounting statistics, for inbound ACLs, for a specific LAG.

```
device# clear access-list accounting lag 1 in
```

The following example clears ACL accounting statistics, for outbound IPv4 ACLs, for a specific LAG.

```
device# clear access-list accounting lag 1 out IPv4
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |
| 08.0.70 | This command was modified to allow clearing of statistics of outbound ACLs. |

# clear acl-on-arp

Clears the count of how many ARP packets have been dropped on the interface.

## Syntax

**clear acl-on-arp**

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

The Filter Count column in the output of the **show acl-on-arp** command shows how many ARP packets have been dropped on the interface since the last time the count was cleared. The **clear acl-on-arp** command resets the filter count on all interfaces in a device back to zero.

## Examples

The following example clears the count of how many ARP packets have been dropped on the interface.

```
device# clear acl-on-arp
```

# clear authentication sessions

Clears 801.1X and MAC authentication sessions on an interface or a range of interfaces.

## Syntax

**clear authentication sessions** [ **ethernet** { *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ] } | **unit** *unit_number* | *mac_address* ]

## Parameters

**ethernet** { *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ] }
Specifies the interface or range of interfaces on which authentication sessions are cleared.

**unit** *unit_number*
Specifies the stack unit on which authentication sessions are cleared.

*mac_address*
Specifies the MAC address (in HHHH.HHHH.HHH form) for which authentication sessions are cleared.

## Modes

Privileged EXEC mode or any configuration mode.

## Usage Guidelines

The **clear authentication sessions** command with no parameters clears sessions for a stack or standalone unit.

## Examples

The following example clears authentication sessions for port 1//1/1.

```
device# clear authentication sessions ethernet 1/1/1
```

The following example clears authentication sessions on a range of ports on unit 1.

```
device# clear authentication sessions ethernet 1/1/1 to 1/1/10
```

The following example clears authentication sessions on stack unit 3.

```
device# clear authentication sessions unit 3
```

The following example clears authentication sessions for an entire stack or for a standalone unit.

```
device# clear authentication sessions
```

The following example clears authentication sessions for the MAC address specified.

```
clear authentication sessions 0000.0034.abd4
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# clear authentication statistics

Clears 802.1X and MAC authentication sessions and statistics on an interface or range of interfaces.

## Syntax

**clear authentication statistics** [ **ethernet** { *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ] } | **unit** *unit_number* ]

## Parameters

**ethernet** { *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ] }

Specifies the interface or range of interfaces on which statistics are cleared.

**unit** *unit_number*

Specifies the stack unit on which statistics are cleared.

## Modes

Privileged EXEC mode or any configuration mode.

## Usage Guidelines

The command **clear authentication statistics** without parameters clears authentication statistics for the entire stack or standalone unit.

## Examples

The following example clears authentication statistic counters for port 1//1/1.

```
device# clear authentication statistics ethernet 1/1/1
```

The following example clears authentication statistics on a range of ports on unit 1.

```
device# clear authentication statistics ethernet 1/1/1 to 1/1/10
```

The following example clears statistics on stack unit 3.

```
device# clear authentication statistics unit 3
```

The following example clears statistics for an entire stack or for a standalone unit.

```
device# clear authentication statistics
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# clear cable diagnostics tdr

Clears the results of Virtual Cable Test (VCT) TDR testing (if any) conducted on the specified port

## Syntax

**clear cable-diagnostics tdr** *stackid**/**slot**/**port*

## Command Default

By default, the results of the previous test (if any) are present and are displayed in response to the **show cable-diagnostics tdr** command for the specified port.

## Parameters

*stackid**/**slot**/**port*

Identifies the specific interface (port), by device, slot, and port number in the format shown.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to clear TDR test registers before every TDR cable diagnostic test. Most ICX devices support VCT technology. VCT technology enables the diagnosis of a conductor (wire or cable) by sending a pulsed signal into the conductor, then examining the reflection of that pulse. This method of cable analysis is referred to as Time Domain Reflectometry (TDR). By examining the reflection, the ICX device can detect and report cable statistics such as local and remote link pair, cable length, and link status.

Use the command in conjunction with the **phy cable-diagnostics tdr stackid/slot/port** command to test the interface.

Show diagnostic test results using the **show cable-diagnostics tdr stackid/slot/port** command.

## Examples

In the following example, results from the previous test are cleared from the third interface on the second slot of the first device in the stack.

```
device# clear cable-diagnostics tdr 1/2/3
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# clear cli-command-history

Clears the allocated logging memory and removes the command log history.

## Syntax

**clear cli-command-history**

## Modes

Privileged EXEC mode

Global configuration mode

## Examples

The following example clears the command log history.

```
device(config)# clear cli-command-history
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# clear dhcp

Clears the DHCP binding database.

## Syntax

**clear dhcp** *ip-address*

## Parameters

*ip-address*
The IP address of the client.

## Modes

User EXEC mode

## Usage Guidelines

You can remove all entries in the database or remove entries for a specific IP address only.

## Examples

The following example removes all entries from the DHCP binding database.

```
device# clear dhcp
```

The following example clears entries for a specific IP address.

```
device# clear dhcp 10.10.102.4
```

# clear dot1x sessions

Clears 802.1X authentication sessions.

## Syntax

**clear dot1x sessions** { *mac-address* | **stack-unit** *id* | **ethernet** *unit*/*slot*/*port* }

## Parameters

*mac-address*
Specifies the MAC address from which the 802.1X authentication sessions are to be cleared.

**stack-unit** *id*
Specifies the stack unit from which the 802.1X authentication sessions are to be cleared.

**ethernet** *unit*/*slot*/*port*
Specifies the interface from which the 802.1X authentication sessions are to be cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to clear the 802.1X authentication sessions.

## Examples

The following example clears the 802.1X authentication session for the specified MAC address.

```
device(config)# clear dot1x sessions 0000.0034.abd4
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.70 | The command was modified to include the stack-unit option. |

# clear dot1x statistics

Clears 802.1X authentication statistics.

## Syntax

**clear dot1x statistics** [ **ethernet** *unit/slot/port* | **stack-unit** *id* ]

## Parameters

**ethernet** *unit/slot/port*

Specifies the interface on which the 802.1X authentication statistics are to be cleared.

**stack-unit** *id*

Specifies the stack unit on which the 802.1X authentication statistics are to be cleared.

## Modes

Privileged EXEC mode

## Examples

The following example clears 802.1X authentication statistics.

```
device(config)# clear dot1x statistics
```

The following example clears 802.1X authentication statistics on a specific interface.

```
device(config)# clear dot1x statistics ethernet 1/1/1
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | The **all** option was removed from ICX 6430, ICX 6450, ICX 6610, FCX, and ICX 7750. |
| 08.0.40 | The **all** option was removed as it is not supported on ICX 7750, ICX 7450, and ICX 7250. |
| 08.0.70 | The **stack-unit** option was added. |

# clear dot1x-mka statistics

Clears current MACsec Key Agreement (MKA) statistics.

## Syntax

**clear dot1x-mka statistics ethernet** *device**/**slot**/**port*

## Parameters

**ethernet** *device**/**slot**/**port*

Specifies an Ethernet interface by device position in stack, slot on the device, and interface on the slot.

## Modes

User EXEC mode

## Usage Guidelines

MACsec commands are supported only on the ICX 7450.

## Examples

In the following example, MKA statistics are cleared for Ethernet interface 1/3/3 (port 3 of slot 3 on the first device in the stack).

```
device# clear dot1x-mka statistics ethernet 1/3/3
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450. |

# clear fdp counters

Clears Foundry Discovery Protocol (FDP) and Cisco Discovery Protocol (CDP) statistics.

## Syntax

**clear fdp counters**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example clears FDP and CDP statistics.

```
device(config)# clear fdp counters
```

# clear fdp table

Clears the information received in Foundry Discovery Protocol (FDP) and Cisco Discovery Protocol (CDP) updates from neighboring devices.

## Syntax

**clear fdp table**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

This command clears all updates for FDP and CDP.

## Examples

The following example clears FDP and CDP updates from neighboring devices.

```
device(config)# clear fdp table
```

# clear gvrp statistics

Clears statistics of the GVRP counters.

## Syntax

**clear gvrp statistics** { **all** | **ethernet** *unit/slot/port* | **lag** *lag-id* }

## Parameters

**all**

Clears the counters for all ports.

**ethernet** *unit/slot/port*

Clears the counters for a specific Ethernet port.

**lag** *lag-id*

Specifies the LAG virtual interface.

## Modes

Privileged EXEC mode

Global configuration mode

GVRP configuration mode

## Examples

The following example shows how to clear statistics for all GVRP counters.

```
device# clear gvrp statistics all
```

The following example shows how to clear statistics for a specific port.

```
device# clear gvrp statistics ethernet 1/2/1
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear ikev2 sa

Clears Internet Key Exchange version 2 security associations (IKEv2 SAs).

## Syntax

**clear ikev2 sa** [ **fvrf** *vrf-name* | **ipv4** | **local** *ip-address* | **remote** *ip-address* ]

## Parameters

**fvrf** *vrf-name*
Specifies the front-door VRF (FVRF) for the SAs.

**ipv4**
Specifies clearing IPv4 connections.

**local** *ip-address*
Specifies the local IPv4 address for the SAs.

**remote** *ip-address*
Specifies the remote IPv4 address for the SAs.

## Modes

Privileged EXEC mode

## Usage Guidelines

The clearing process deletes and re-establishes the SAs (including any child SAs).

When optional parameters are not specified, the command clears all IKEv2 SAs on the device.

**NOTE**
Clearing all IKEv2 SAs is a costly operation. Therefore, the unqualified version of the command should be used with caution. Issuing multiple unqualified versions of the command within a short time frame is not recommended.

## Examples

The following example clears the IKEv2 SAs for local interface 10.10.20.1.

```
device# clear ikev2 sa local 10.10.20.1
```

The following example clears the IKE SAs for remote interface 10.0.10.1.

```
device# clear ikev2 sa remote 10.0.10.1
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# clear ikev2 statistics

Clears Internet Key Exchange version 2 (IKEv2) statistics by resetting the various IKEv2 counters to zero.

## Syntax

**clear ikev2 statistics**

## Modes

Privileged EXEC mode

## Examples

The following example clears IKEv2 statistics from the device.

```
device# clear ikev2 statistics
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# clear ip bgp dampening

Reactivates suppressed BGP4 routes.

## Syntax

**clear ip bgp dampening** [ *ip-addr* { **/** *mask* } ]

## Parameters

*ip-addr*

IPv4 address of a specified route in dotted-decimal notation.

*mask*

IPv4 mask of a specified route in CIDR notation.

## Modes

Privileged EXEC mode

## Examples

The following example unsuppresses all suppressed BGP4 routes.

```
device# clear ip bgp dampening
```

# clear ip bgp flap-statistics

Clears the dampening statistics for a BGP4 route without changing the dampening status of the route.

## Syntax

**clear ip bgp flap-statistics** [ *ip-addr* { **/** *mask* } | **neighbor** *ip-addr* | **regular-expression** *string* ]

## Parameters

*ip-addr*
: Specifies the IPv4 address of a specified route in dotted-decimal notation.

*mask*
: Specifies the IPv4 mask of a specified route in CIDR notation.

**neighbor**
: Clears dampening statistics only for routes learned from the specified neighbor.

*ip-addr*
: Specifies the IPv4 address of the neighbor.

**regular-expression**
: Specifies a regular expression.

*string*
: Regular expression.

## Modes

Privileged EXEC mode

## Examples

The following example clears the dampening statistics for a BGP4 route.

```
device# clear ip bgp flap-statistics 10.0.0.0/16
```

# clear ip bgp local routes

Clears BGP4 local routes from the IP route table and resets the routes.

## Syntax

**clear ip bgp local routes**

## Modes

Privileged EXEC mode

## Examples

The following example clears all BGP4 local routes.

```
device# clear ip bgp local routes
```

# clear ip bgp neighbor

Requests a dynamic refresh of BGP4 connections or routes from a neighbor, with a variety of options.

## Syntax

**clear ip bgp neighbor** { **all** | *as-num* | *peer-group-name* | *ip-addr* } [ **last-packet-with-error** ] [ **notification-errors** ] [ **soft** [ **in** | **out** ] ] [ **soft-outbound** ] [**traffic** ]

## Parameters

**all**

Resets and clears all BGP4 connections to all neighbors.

*as-num*

Clears all BGP4 connections within this autonomous system. Range is from 1 through 4294967295.

*peer-group-name*

Clears all BGP4 connections in this peer group. Range is from 1 through 63 characters.

*ip-addr*

Clears all BGP4 connections with this IPv4 address, in dotted-decimal notation.

**last-packet-with-error**

Clears all BGP4 connections identified as having the last packet received with an error.

**notification-errors**

Clears all BGP4 connections identified as having notification errors.

**soft**

Refreshes routes received from or sent to the neighbor.

**in**

Refreshes received routes.

**out**

Refreshes sent routes.

**soft-outbound**

Refreshes all outbound routes by applying new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

**NOTE**

Use **soft-outbound** only if the outbound policy is changed. This operand updates all outbound routes by applying the new or changed filters. However, the device sends to the neighbor only the existing routes that are affected by the new or changed filters. The **soft out** operand updates all outbound routes and then sends the entire BGP4 route table on the device to the neighbor after the device changes or excludes the routes affected by the filters.

**traffic**

Clears the counters (resets them to 0) for BGP4 messages.

# Modes

Privileged EXEC mode

# Examples

The following example refreshes all BGP4 neighbor connections.

```
device# clear ip bgp neighbor all
```

# clear ip bgp routes

Clears BGP4 routes from the IP route table and resets the routes.

## Syntax

**clear ip bgp routes** [ *ip-addr* [ **/** *mask* ] ]

## Parameters

*ip-addr*

Specifies the IPv4 address of a specified route in dotted-decimal notation.

*mask*

Specifies the IPv4 mask of a specified route in CIDR notation.

## Modes

Privileged EXEC mode

## Examples

The following example clears all BGP4 routes.

```
device# clear ip bgp routes 10.0.0.0/16
```

# clear ip bgp traffic

Clears the BGP4 message counter for all neighbors.

## Syntax

**clear ip bgp traffic**

## Modes

Privileged EXEC mode

## Examples

The following example clears the BGP4 message counters:

```
device# clear ip bgp traffic
```

# clear ip bgp vrf

Clears BGP4 information for a virtual routing and forwarding (VRF) instance.

## Syntax

**clear ip bgp vrf** *vrf-name*

## Parameters

**vrf** *vrf-name*

Specifies the name of a VRF instance.

## Modes

Privileged EXEC mode

## Examples

The following example clears BGP4 information for VRF red.

```
device# clear ip bgp vrf red
```

# clear ip dhcp-server binding

Clears the leases from the lease binding database.

## Syntax

**clear ip dhcp-server binding** { *address* | ***** }

## Parameters

*address*

The IP address to be deleted.

*****

Wildcard clears all lease entries.

## Modes

Global configuration mode.

## Usage Guidelines

Use this command to delete to delete a specific lease, or all lease entries from the lease binding database.

## Examples

The following example clears all lease entries.

```
device(config)# clear ip dhcp-server binding *
```

# clear ip dhcp-server statistics

Resets all DHCP server packet statistics, or server packet statistics for a specified pool.

## Syntax

**clear ip dhcp-server statistics** [ *pool-name* ]

## Parameters

*pool-name*

Specifies a pool in ASCII characters.

## Modes

Privileged EXEC mode.

## Usage Guidelines

The **show ip dhcp-server summary** command displays packet counters that are received to the DHCP server for a specified pool or all pools. DHCP must be enabled before this command can be executed.

## Examples

The following example resets all DHCP server packet statistics.

```
device# clear ip dhcp-server statistics
```

The following example resets DHCP server packet statistics for a specified pool.

```
device# clear ip dhcp-server statistics poola
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# clear ip igmp cache

Clears the IGMP group membership table from a VRF instance or from all interfaces on the device.

## Syntax

**clear ip igmp** [ **vrf** *vrf-name* ] **cache**

## Parameters

**vrf** *vrf-name*

Specifies the name of a VRF instance. When this parameter is specified, the command is executed only on the specified VRF instance.

**cache**

Clears the IGMP group membership table from a specified VRF instance or from all interfaces.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears information for all VRF instances.

## Examples

The following example clears the IGMP group membership table for the device.

```
device# clear ip igmp cache
```

The following example clears the IGMP membership information on a single virtual routing interface, vpn1.

```
device# clear ip igmp vrf vpn1 cache
```

# clear ip igmp traffic

Clears statistics for IGMP traffic from a VRF instance or from all interfaces on the device.

## Syntax

**clear ip igmp** [ **vrf** *vrf-name* ] **traffic**

## Parameters

**vrf** *vrf-name*

Specifies the name of a VRF instance. When this parameter is specified, the command is executed only on the specified VRF instance.

**traffic**

Clears multicast traffic statistics from a specified VRF instance or from all interfaces.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears information for all VRF instances.

## Examples

The following example clears all multicast statistics on the device.

```
device# clear ip igmp traffic
```

The following example clears the multicast statistics on the virtual routing interface, vpn1.

```
device# clear ip igmp vrf vpn1 traffic
```

# clear ip mroute

Removes multicast routes from the IP multicast routing table .

## Syntax

**clear ip mroute** [ **vrf** *vrf-name* ] [ *ip-address* {*ip-mask* | *mask-bits* } ]

## Parameters

**vrf** *vrf-name*
: Specifies a VRF.

*ip-address*
: Specifies an IP address.

*ip-mask*
: Specifies an IP subnet mask.

*mask-bits*
: Specifies a subnet mask in bits.

## Modes

Global configuration mode

## Usage Guidelines

After multicast routes are cleared from an IP multicast routing table, the best static multicast routes are added back to the routing table.

When used without specifying a **vrf** *vrf-name* this command clears multicast routes from the multicast routing table.

## Examples

The following example removes all mroutes from the IP multicast routing table:

```
Device# configure terminal
Device(config)# clear ip mroute
```

The following example removes all mroutes from the vrf green IP multicast routing table:

```
Device# configure terminal
Device(config)# clear ip mroute vrf green
```

The following example removes mroute 10.0.0.2/24 from the IP multicast routing table:

```
Device# configure terminal
Device(config)# clear ip mroute 10.0.0.2/24
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# clear ip msdp peer

Clears multicast source discovery protocol (MSDP) peer information.

## Syntax

**clear ip msdp** [ **vrf** *vrf-name* ] **peer** [ *ip-addr* ]

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**peer**
Clears MSDP peer information.

*ip-addr*
Specifies a VRF peer. If you do not specify a peer, MSDP information for all peers is cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears information for all VRF instances.

A message is displayed when the connection is closed.

## Examples

The following example clears the MSDP peer connection with MSDP router 192.168.162.1.

```
device# clear ip msdp peer 192.168.162.1
```

# clear ip msdp sa-cache

Clears the multicast source discovery protocol (MSDP) source active (SA) cache.

## Syntax

**clear ip msdp** [ **vrf** *vrf-name* ] **sa-cache** [ *ip-addr* ]

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**sa-cache**
Clears MSDP source active cache information.

*ip-addr*
Specifies a source or a group to clear. If you do not specify a source or group, all SA cache entries are cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears the SA cache for all VRF instances.

## Examples

The following example clears the MSDP SA cache.

```
device# clear ip msdp sa-cache
```

# clear ip msdp statistics

Clears multicast source discovery protocol (MSDP) statistics.

## Syntax

**clear ip msdp** [ **vrf** *vrf-name* ] **statistics** [ *ip-addr* ]

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**statistics**
Clears MSDP statistics information.

*ip-addr*
Specifies a VRF peer.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears information for all VRF instances.

## Examples

The following example clears MSDP statistics.

```
device# clear ip msdp statistics
```

# clear ip multicast counters

Clears IGMP snooping on error and traffic counters for all VLANs.

## Syntax

**clear ip multicast counters**

## Modes

Privileged EXEC mode

## Examples

The following example clears IGMP snooping on error and traffic counters for all VLANs.

```
device# clear ip multicast counters
```

# clear ip multicast mcache

Clears the mcache on a specific VLAN or on all VLANs.

## Syntax

**clear ip multicast** [ **vlan** *vlan-id* ] **mcache** [ *ip-addr* ]

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

**mcache**

Clears the mcache on the specified VLANs.

*ip-addr*

Specifies a source or a group to clear. If you do not specify a source or group, all cache entries are cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vlan** keyword, this command clears the mcache for all VLANs.

## Examples

The following example clears the mcache on all VLANs:

```
device# clear ip multicast mcache
```

The following example clears the mcache on VLAN 20.

```
device# clear ip multicast vlan 20 mcache
```

# clear ip multicast traffic

Clears traffic counters on a specific VLAN or on all VLANs.

## Syntax

**clear ip multicast** [ **vlan** *vlan-id* ] **traffic**

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

**traffic**

Clears traffic counters on the specified VLANs.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vlan** keyword, this command clears information for all VLANs.

## Examples

The following example clears traffic counters on VLAN 20.

```
device# clear ip multicast vlan 20 traffic
```

The following example clears the traffic counters on all on VLANs.

```
device# clear ip multicast traffic
```

# clear ip ospf

Clears OSPF process, counters, neighbors, or routes.

## Syntax

**clear ip ospf all**

**clear ip ospf neighbor** { *A.B.C.D* | **all** } [ **ethernet** *unit/slot/port* | **lag** *lag-id* | **tunnel** *number* | **ve** *vlan_id* ]

**clear ip ospf routes** { *A.B.C.D/L* | **all** }

**clear ip ospf traffic**

## Parameters

**all**

Globally resets (disables then re-enables) OSPF without deleting the OSPF configuration information.

**neighbor**

Clears the specified neighbor, or clears all neighbors.

*A.B.C.D*

Specifies the IP address of the neighbor to clear.

**all**

Clears all neighbors.

**ethernet** *unit/slot/port*

Specifies the Ethernet interface and the interface ID in the format unit/slot/port.

**lag** *lag-id*

Specifies the LAG virtual interface.

**tunnel** *number*

Specifies a tunnel.

**ve** *vlan_id*

Specifies a virtual Ethernet (VE) interface.

**routes**

Clears matching routes or clears all routes.

*A.B.C.D*

Clears all routes that match the prefix and mask that you specify.

**all**

Clears all routes.

**traffic**

Clears OSPF counters and errors.

## Modes

User EXEC mode

# Examples

The following example resets OSPF without deleting the OSPF configuration.

```
device# clear ip ospf all
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear ip pim cache

Clears the PIM forwarding cache on a specific VRF instance or on all VRFs.

## Syntax

**clear ip pim** [ **vrf** *vrf-name* ] **cache** [ *ip-address* ]

## Parameters

**vrf** *vrf-name*
: Specifies a VRF instance.

**cache**
: Specifies the PIM forwarding cache.

*ip-address*
: Specifies the source or group address of the entry to clear.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears the PIM forwarding cache for all VRFs.

## Examples

The following example clears the PIM forwarding cache on a VRF instance named blue.

```
device# clear ip pim vrf blue cache
```

# clear ip pim counters

Clears PIM message counters.

## Syntax

**clear ip pim** [ **vrf** *vrf-name* ] **counters**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**counters**
Specifies PIM message counters.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears the PIM message counters for all VRFs.

## Examples

The following example clears the PIM message counters.

```
Device# clear ip pim counters
```

The following example clears the PIM message counters on a XRF named blue.

```
Device# clear ip pim vrf blue counters
```

# clear ip pim hw-resource

Clears the PIM hardware resource fail count for a specific VRF instance or for all VRFs.

## Syntax

**clear ip pim** [ **vrf** *vrf-name* ] **hw-resource**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**hw-resource**
Specifies hardware resource fail count.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears the PIM hardware resource fail count for all VRFs.

## Examples

The following example clears the PIM hardware resource fail count.

```
Device# clear ip pim hw-resource
```

# clear ip pim rp-map

Updates the entries in the static multicast forwarding table for a specific VRF instance or for all VRFs.

## Syntax

**clear ip pim** [ **vrf** *vrf-name* ] **rp-map**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**rp-map**
Specifies the entries in a PIM sparse static multicast forwarding table.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears the PIM forwarding cache for all VRFs.

Configure this command to update the entries in the static multicast forwarding table immediately after making rendezvous point (RP) configuration changes. This command is meant to be used with the **rp-address** command.

## Examples

The following example clears the entries in a PIM sparse static multicast forwarding table on a VRF instance named blue.

```
Device# clear ip pim vrf blue rp-map
```

# clear ip pim traffic

Clears PIM traffic for a specific VRF instance or on all VRFs.

## Syntax

**clear ip pim** [ **vrf** *vrf-name* ] **traffic**

## Parameters

**vrf** *vrf-name*
: Specifies a VRF instance.

**traffic**
: Specifies PIM traffic.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears PIM traffic for all VRFs.

## Examples

The following example clears PIM traffic on a VRF instance named blue.

```
device# clear ip pim vrf blue traffic
```

# clear ip pimsm-snoop

Clears PIM sparse mode (SM) information.

## Syntax

**clear ip pimsm-snoop** [ **vlan***vlan-id* ] { **cache** [ *ip-address* ] | **stats**}

## Parameters

**vlan***vlan-id*

Specifies clearing information on a specific VLAN.

**cache**

Specifies clearing the PIM SM snooping cache.

*ip-address*

Specifies clearing PIM SM snooping-cache information on a specific source or group.

**stats**

Specifies clearing traffic and error counters.

## Modes

Global configuration mode

## Examples

The following example clears PIM SM information from all VLANs.

```
Device(config)#clear ip pimsm-snoop cache
```

The following example clears PIM SM information from a specific VLAN.

```
Device(config)#clear ip pimsm-snoop vlan 10 cache
```

The following example clears PIM SM information from a specific source.

```
Device(config)#clear ip pimsm-snoop cache 10.1.1.1
```

The following example clears traffic and error counters from all VLANs.

```
Device(config)#clear ip pimsm-snoop stats
```

## History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was introduced. |

# clear ip route

Clears entire IP route table or specific routes.

## Syntax

**clear ip route** [ **vrf** *vrf-name* ] [ *ip-address* ]

## Parameters

**vrf** *vrf-name*

Specifies the VPN Routing and Forwarding instance.

*ip-address*

Specifies the route entry to be cleared from the IP route table. The IP address can be specified in the format A.B.C.D/L where L is the mask bits or as A.B.C.D followed by network mask.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command, when used without any parameters, clears the entire IP route table.

When an interface subnet route with an interface address that directly matches a host route learned from a neighboring device is configured and subsequently removed, the **clear ip route** command should be used so that the learned route is updated in the Routing and Hardware Forwarding table.

**NOTE**
The L2 and L3 protocols might flap in case the number of L3 routes are more.

## Examples

The following example clears the IP route 10.157.22.0/24 from the IP routing table.

```
device# clear ip route 10.157.22.0/24
```

# clear ip tunnel

Clears statistics (reset all fields to zero) for all IP tunnels or for a specific tunnel interface.

## Syntax

**clear ip tunnel** { **pmtud** *tunnel-id* | **stat** [ *tunnel-id* ]}

## Parameters

**pmtud** *tunnel-id*

Resets a dynamically-configured MTU on a tunnel Interface back to the configured value.

**stat**

Clears statistics of all tunnels.

*tunnel-id*

Clears statistics of the specified tunnel.

## Modes

Privileged EXEC mode

## Usage Guidelines

You can also use the **clear statistics tunnel** command to clear tunnel statistics.

## Examples

The following example clears statistics for all IP tunnels.

```
device# clear ip tunnel stat
```

The following example clears the statistics for a specific tunnel interface.

```
device# clear ip tunnel stat 2
```

The following example resets a dynamically-configured MTU on a tunnel interface.

```
device# clear ip tunnel pmtud 1
```

# clear ip vrrp statistics

Clears IPv4 Virtual Router Redundancy Protocol (VRRP) statistics.

## Syntax

**clear ip vrrp statistics**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be entered in privileged EXEC mode and in any configuration mode. Entering the command in a configuration mode can be useful if you are configuring VRRP options, for example, and want to clear existing statistics.

## Examples

The following example clears IPv4 VRRP statistics when entered in privileged EXEC mode.

```
device# clear ip vrrp statistics
```

The following example clears IPv4 VRRP statistics when entered in VRID interface configuration mode.

```
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# clear ip vrrp statistics
```

# clear ip vrrp-extended statistics

Clears IPv4 Virtual Router Redundancy Protocol (VRRP) Extended (VRRP-E) statistics.

## Syntax

**clear ip vrrp-extended statistics**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be entered in privileged EXEC mode and in any configuration mode. Entering the command in a configuration mode can be useful if you are configuring VRRP-E options, for example, and want to clear existing statistics.

## Examples

The following example clears IPv4 VRRP-E statistics when entered in privileged EXEC mode.

```
device# clear ip vrrp-extended statistics
```

The following example clears IPv4 VRRP-E statistics when entered in VRID interface configuration mode.

```
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.4.1/24
device(config-if-e1000-1/1/5)# ip vrrp-extended vrid 2
device(config-if-e1000-1/1/5-vrid-2)# clear ip vrrp-extended statistics
```

# clear ipsec sa

Clears IPsec security associations (SAs).

## Syntax

**clear ipsec sa** [ **fvrf** *vrf-name* | **ipv4** | **peer** *ip-address* ]

## Parameters

**fvrf** *vrf-name*
Specifies the front-door VRF (FVRF) for the SAs.

**ipv4**
Specifies clearing IPv4 associations.

**peer** *ip-address*
Specifies clearing associations for the IPv4 address of a peer.

## Modes

Privileged EXEC mode

## Usage Guidelines

The clearing process deletes and re-establishes IPsec SAs. The SAs remain unchanged.

When optional parameters are not specified, this command clears all IPsec SAs on the device.

**NOTE**
Clearing all IPsec SAs is a costly operation. Therefore, the unqualified version of the command should be used with caution. Issuing multiple unqualified versions of the command within a short time frame is not recommended.

## Examples

The following example clears all IPsec SAs on the device.

```
device# clear ipsec sa
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# clear ipv6 bgp dampening

Reactivates suppressed BGP4+ routes.

## Syntax

**clear ipv6 bgp dampening** [ *ipv6-addr* { **/** *mask* } ]

## Parameters

*ipv6-addr*

IPv6 address of a specified route in dotted-decimal notation.

*mask*

IPv6mask of a specified route in CIDR notation.

## Modes

Privileged EXEC mode

## Examples

The following example unsuppresses all suppressed BGP4+ routes.

```
device# clear ipv6 bgp dampening
```

# clear ipv6 bgp flap-statistics

Clears the dampening statistics for a BGP4+ route without changing the dampening status of the route.

## Syntax

**clear ipv6 bgp flap-statistics** [ *ipv6-addr* { **/** *mask* } | **neighbor** *ipv6-addr* | **regular-expression** *string* ]

## Parameters

*ipv6-addr*

Specifies the IPv6 address of a specified route in dotted-decimal notation.

*mask*

Specifies the IPv6 mask of a specified route in CIDR notation.

**neighbor**

Clears dampening statistics only for routes learned from the specified neighbor.

*ipv6-addr*

Specifies the IPv6 address of the neighbor.

**regular-expression**

Specifies a regular expression.

*string*

Regular expression.

## Modes

Privileged EXEC mode

## Examples

The following example clears the dampening statistics for a BGP4+ route.

```
device# clear ipv6 bgp flap-statistics 2001:2002::23:61
```

# clear ipv6 bgp local routes

Clears BGP4+ local routes from the IP route table and resets the routes.

## Syntax

**clear ipv6 bgp local routes**

## Modes

Privileged EXEC mode

## Examples

The following example clears all BGP4+ local routes.

```
device> clear ipv6 bgp local routes
```

# clear ipv6 bgp neighbor

Requests a dynamic refresh of BGP4+ connections or routes from a neighbor, with a variety of options.

## Syntax

**clear ipv6 bgp neighbor** { **all** | *as-num* | *peer-group-name* | *iipv6-addr* } [ **last-packet-with-error** ] [ **notification-errors** ] [ **soft** [ **in** | **out** ] ] [ **soft-outbound** ] [**traffic** ]

## Parameters

**all**

Resets and clears all BGP4+ connections to all neighbors.

*as-num*

Clears all BGP4+ connections within this autonomous system. Range is from 1 through 4294967295.

*peer-group-name*

Clears all BGP4+ connections in this peer group. Range is from 1 through 63 characters.

*ipv6-addr*

Clears all BGP4+ connections with this IPv6 address, in dotted-decimal notation.

**last-packet-with-error**

Clears all BGP4+ connections identified as having the last packet received with an error.

**notification-errors**

Clears all BGP4+ connections identified as having notification errors.

**soft**

Refreshes routes received from or sent to the neighbor.

**in**

Refreshes received routes.

**out**

Refreshes sent routes.

**soft-outbound**

Refreshes all outbound routes by applying new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

**NOTE**

Use **soft-outbound** only if the outbound policy is changed. This operand updates all outbound routes by applying the new or changed filters. However, the device sends to the neighbor only the existing routes that are affected by the new or changed filters. The **soft out** operand updates all outbound routes and then sends the entire BGP4+ route table on the device to the neighbor after the device changes or excludes the routes affected by the filters.

**traffic**

Clears the counters (resets them to 0) for BGP4+ messages.

# Modes

Privileged EXEC mode

# Examples

The following example refreshes all BGP4+ neighbor connections.

```
device# clear ipv6 bgp neighbor all
```

# clear ipv6 bgp routes

Clears BGP4+ routes from the route table and resets the routes.

## Syntax

**clear ipv6 bgp routes** [ *ipv6-addr* { **/** *mask* } ]

## Parameters

*ipv6-addr*

Specifies the IPv6 address of a specified route in dotted-decimal notation.

*mask*

Specifies the IPv6 mask of a specified route in CIDR notation.

## Modes

Privileged EXEC mode

## Examples

The following example clears all BGP4+ routes.

```
device> clear ipv6 bgp routes
```

# clear ipv6 bgp traffic

Clears the BGP4+ message counter for all neighbors.

## Syntax

**clear ipv6 bgp traffic**

## Modes

Privileged EXEC mode

## Examples

The following example clears the BGP4+ message counters.

```
device# clear ipv6 bgp traffic
```

# clear ipv6 cache

Deletes all entries in the dynamic host IPv6 cache.

## Syntax

**clear ipv6 cache** [ **vrf** *vrf-name* ] [ *ipv6-address* | *ipv6-prefix*/*prefix-length* | **ethernet** *unit/slot/port* | **lag** *lag-id* | **tunnel** *tunnel-id* | **ve** *ve-number* ]

## Parameters

**vrf** *vrf-name*
Removes cache entries for the specified VPN Routing/Forwarding (VRF) instance.

*ipv6-address*
Removes cache entries for the specified IPv6 address.

*ipv6-prefix*/*prefix-length*
Removes cache entries for the specified IPv6 prefix.

**ethernet** *unit/slot/port*
Removes cache entries for the specified Ethernet interface.

**tunnel** *tunnel-id*
Removes cache entries for the specified tunnel interface.

**lag** *lag-id*
Specifies the LAG virtual interface.

**ve** *ve-number*
Removes cache entries for the specified Virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

You can remove all entries from the IPv6 cache or specify an entry based on the IPv6 prefix, IPv6 address, or interface type.

You must specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

You must specify the *ipv6-address* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

# Examples

The following example removes entries for IPv6 address 2000:e0ff::1.

```
device# clear ipv6 cache 2000:e0ff::1
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear ipv6 dhcp6 snooping

Clears the IPv6 DHCP snooping database.

## Syntax

**clear ipv6 dhcp6 snooping** *vlan*

## Parameters

*vlan*

Specifies the VLAN.

## Modes

Global configuration mode

User EXEC mode

## Usage Guidelines

You can remove all entries in the database, or remove entries for a specific IP address only.

## Examples

The following command clears the IPv6 entries in the database.

```
device# clear ipv6 dhcp6 snooping
```

# clear ipv6 dhcp-relay delegated-prefixes

Clears the IPv6 DHCP relay delegated prefixes.

## Syntax

**clear ipv6 dhcp-relay delegated-prefixes** { **vrf** *vrf-name* | **X:X::X:X/M** | **all** | **interface** *interface-id* }

## Parameters

**vrf** *vrf-name*

Clears the DHCPv6 delegated prefixes for a specific VRF. If this parameter is not provided, then the information for the default VRF is cleared

**X:X::X:X/M**

Clears the specified delegated prefix and removes the corresponding route permanently from the router.

**all**

Clear all the delegated prefixes and remove the corresponding routes permanently from the router for the VRF

**interface** *interface-id*

Clears all the delegated prefixes and removes the corresponding routes permanently from the router for the specified outgoing interface.

## Modes

Privileged EXEC mode.

## Examples

The following example clears the IPv6 DHCP relay delegated prefixes from VRF1.

```
device# clear ipv6 dhcp-relay delegated-prefixes vrf VRF1
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# clear ipv6 dhcp-relay statistics

Clears the IPv6 DHCP packet counters.

## Syntax

**clear ipv6 dhcp-relay statistics**

## Modes

Privileged EXEC mode

## Examples

The following example clears the IPv6 DHCP packet counters.

```
device# clear ipv6 dhcp-relay statistics
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# clear ipv6 mld traffic

Clears the counters on IPv6 multicast listening discovery (MLD) traffic.

## Syntax

**clear ipv6 mld** [ **vrf** *vrf-name* ] **traffic**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**traffic**
Clears the traffic counters.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears traffic counters for all VRF instances.

## Examples

The following example clears counters on IPv6 PIM traffic.

```
device# clear ipv6 mld traffic
```

# clear ipv6 mroute

Removes IPv6 multicast routes from the IPv6 multicast routing table.

## Syntax

**clear ipv6 mroute** [ **vrf** *vrf-name*] [ *ipv6-address-prefix/prefix-length* ]

## Parameters

**vrf** *vrf-name*

Specifies a VRF route.

*ipv6-address-prefix/prefix-length*

Specifies an IPv6 address prefix in hexadecimal using 16-bit values between colons as documented in RFC 2373 and a prefix length as a decimal value.

## Modes

Privileged EXEC mode

## Usage Guidelines

After mroutes are removed from an IPv6 multicast routing table, the best static mroutes are added back to it.

## Examples

The following example removes all mroutes from the IPv6 multicast routing table:

```
Device(config)# clear ipv6 mroute
```

The following example removes all mroutes from the vrf green IPv6 multicast routing table:

```
Device(config)# clear ipv6 mroute vrf green
```

The following example removes mroute 2000:7838::/32 from the IPv6 multicast routing table:

```
Device(config)# clear ipv6 mroute 2000:7838::/32
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# clear ipv6 multicast counters

Clears multicast listening discovery (MLD) snooping error and traffic counters on all VLANs.

## Syntax

**clear ipv6 multicast counters**

## Modes

Privileged EXEC mode

## Examples

The following example clears MLD snooping on error and traffic counters for all VLANs.

```
device# clear ipv6 multicast counters
```

# clear ipv6 multicast mcache

Clears the multicast listening discovery (MLD) mcache on a specific VLAN or on all VLANs.

## Syntax

**clear ipv6 multicast** [ **vlan** *vlan-id* ] **mcache** [ *ipv6-addr* ]

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

**mcache**

Clears the mcache on the specified VLANs.

*ipv6-addr*

Specifies a source or a group to clear. If you do not specify a source or group, all cache entries are cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vlan** keyword, this command clears information for all VLANs.

## Examples

The following example clears the mcache on VLAN 20.

```
device# clear ipv6 multicast vlan 20 mcache
```

# clear ipv6 multicast traffic

Clears multicast listening discovery (MLD) traffic counters on a specific VLAN or on all VLANs.

## Syntax

**clear ipv6 multicast** [ **vlan** *vlan-id* ] **traffic**

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

**traffic**

Clears traffic counters on the specified VLANs.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vlan** keyword, this command clears information for all VLANs.

## Examples

The following example clears traffic counters on VLAN 20.

```
device# clear ipv6 multicast vlan 20 traffic
```

# clear ipv6 neighbor

Removes entries from the IPv6 neighbor table.

## Syntax

**clear ipv6 neighbor** [ **vrf** *vrf-name* ] [ *ipv6-address* | *ipv6-prefix***/***prefix-length* | **ve** *ve-number* ]

**clear ipv6 neighbor ethernet** *unit/slot/port*

**clear ipv6 neighbor lag** *lag-id*

## Parameters

**vrf** *vrf-name*
Removes entries from the IPv6 neighbor table for the specified VPN Routing/Forwarding (VRF) instance.

*ipv6-address*
Removes cache entries for the specified IPv6 address.

*ipv6-prefix***/***prefix-length*
Removes cache entries for the specified IPv6 prefix.

**ethernet** *unit/slot/port*
Removes cache entries for the specified Ethernet interface.

**lag** *lag-id*
Specifies the LAG virtual interface.

**ve** *ve-number*
Removes cache entries for the specified Virtual Ethernet (VE) interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

You must specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

You must specify the *ipv6-address* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

## Examples

The following example removes neighbor entries for Ethernet interface 1/3/1.

```
device# clear ipv6 neighbor ethernet 1/3/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear ipv6 neighbor inspection

Clears the static neighbor discovery (ND) inspect entries and ND inspection statistics.

## Syntax

**clear ipv6 neighbor** [ **vrf** *vrf-name* ] **inspection** [ **static-entry** | **statistics** ]

## Parameters

**vrf**

Specifies the VRF instance (optional).

*vrf-name*

Specifies the ID of the VRF instance required with **vrf** .

inspection

Specifies that the neighbor discovery messages are verified against the static ND inspection entries or dynamically learned DHCPv6 snoop entries.

**static-entry**

Clears the manually configured static ND inspect entries that are used to validate the packets received on untrusted ports.

**statistics**

Clears the total number of neighbor discovery messages received and the number of packets discarded after ND inspection.

## Modes

Privileged EXEC mode

Global configuration mode

VRF configuration mode

## Usage Guidelines

This command can be used in three different modes as shown in the examples. If used without specifying a VRF, this command clears data from the default VRF.

## Examples

The following example removes the manually configured static ND inspect entries.

```
device# clear ipv6 neighbor inspection static-entry
```

The following example removes the manually configured static ND inspect entries on a VRF.

```
device# configure terminal
device(config)# vrf vrf2
device(config-vrf-vrf2)# clear ipv6 neighbor vrf vrf2 inspection static-entry
```

The following example deletes the ND inspection statistics.

```
device# configure terminal
device(config)# clear ipv6 neighbor inspection statistics
```

The following example deletes the ND inspection statistics on a VRF.

```
device# configure terminal
device(config)# clear ipv6 neighbor vrf vrf2 inspection statistics
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# clear ipv6 ospf

Clears OSPFv3 data processes, counts, force-spf, neighbors, redistribution, routes, and traffic.

## Syntax

**clear ipv6 ospf all**

**clear ipv6 ospf counts**

**clear ipv6 ospf counts neighbor** *A.B.C.D*

**clear ipv6 ospf counts neighbor interface** { **ethernet** *unit/slot/port* | **lag** *lag-id* | **tunnel** *number* | **ve** *vlan_id* } [ *A.B.C.D* ]

**clear ipv6 ospf** { **force-spf** | **redistribution** | **traffic** } [ **vrf** *vrf-name* ]

**clear ipv6 ospf neighbor all**

**clear ipv6 ospf neighbor interface** { **ethernet** *unit/slot/port* | **lag** *lag-id* | **tunnel** *number* | **ve** *vlan_id* } [ *A.B.C.D* ]

**clear ipv6 ospf routes** { *IPv6addr* | **all** }

## Parameters

**all**
Clears all OSPFv3 data.

**counts**
Clears OSPFv3 counters.

**neighbor**
Clears all OSPF counters for a specified neighbor.

*A.B.C.D*
Specifies a neighbor.

**interface**
Specifies an interface.

**ethernet** *unit/slot/port*
Specifies the Ethernet interface and the interface ID in the format unit/slot/port.

**lag** *lag-id*
Specifies the LAG virtual interface.

**tunnel** *number*
Specifies a tunnel interface.

**ve** *vlan_id*
Specifies a virtual Ethernet (VE) interface.

**force-spf**
Performs the shortest path first (SPF) calculation without clearing the OSPFv3 database.

**redistribution**
Clears OSPFv3 redistributed routes.

**traffic**
Clears OSPFv3 traffic statistics.

**routes**
Clears OSPFv3 routes.

# Modes

Privileged EXEC mode

# Usage Guidelines

Use the **force-spf** keyword to perform the shortest path first (SPF) calculation without clearing the OSPFv3 database.

# Examples

The following example restarts the OSPFv3 processes.

```
device# clear ipv6 ospf all
```

The following example clears all OSPFv3 counters for a specified neighbor.

```
device# clear ipv6 ospf counts neighbor 10.10.10.1
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear ipv6 pim cache

Clears the IPv6 PIM forwarding cache.

## Syntax

**clear ipv6 pim** [ **vrf** *vrf-name* ] **cache** *ipv6-address*

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**cache** *ipv6-address*
Specifies group or address of the PIM forwarding cache to clear.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears information for all VRF instances.

## Examples

This example shows how to clear the IPv6 PIM forwarding cache:

```
Device#clear ipv6 pim cache 2001:0DB8:0:1::1/120 5100::192:1:1:1
```

# clear ipv6 pim counters

Clears IPv6 PIM message counters.

## Syntax

**clear ipv6 pim** [ **vrf** *vrf-name* ] **counters**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**counters**
Specifies the IPv6 PIM message counters.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears information for all VRF instances.

## Examples

Thi example shows how to clear the IPv6 PIM message counters:

```
Device#clear ipv6 pim counters
```

# clear ipv6 pim hw-resource

Clears the IPv6 PIM hardware resource fail count for a specific VRF instance or for all VRFs.

## Syntax

**clear ipv6 pim hw-resource**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**hw-resource**
Specifies hardware resource fail count.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears the PIM hardware resource fail count for all VRFs.

## Examples

The following example clears the IPv6 PIM hardware resource fail count.

```
Device# clear ipv6 pim hw-resource
```

# clear ipv6 pim rp-map

Clears the entries in an IPv6 PIM Sparse static multicast forwarding table, allowing a new rendezvous point (RP) configuration to be effective immediately.

## Syntax

**clear ipv6 pim** [ **vrf** *vrf-name* ] **rp-map**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**rp-map**
Specifies the entries in a PIM sparse static multicast forwarding table.

## Modes

Privileged EXEC mode

## Usage Guidelines

Configuring this command clears and overwrites the static RP configuration. If you change the static RP configuration, the entries in the IPv6 PIM Sparse multicast forwarding table continue to use the old RP configuration until they are aged out. You can configure the **clear ipv6 pim rp-map** command to update the entries in the static multicast forwarding table immediately after making RP configuration changes.

This command is meant to be used with the **rp-address** command.

## Examples

This example shows how to clear the entries in an IPv6 PIM Sparse static multicast forwarding table after you change the RP configuration:

```
Device#clear ipv6 pim rp-map
```

# clear ipv6 pim traffic

Clears counters on IPv6 PIM traffic.

## Syntax

**clear ipv6 pim** [ **vrf** *vrf-name* ] **traffic**

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

**traffic**
Specifies counters on IPv6 PIM traffic.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** keyword, this command clears counters for all VRF instances.

## Examples

This example shows how to clear IPv6 PIM traffic counters on all VRF instances:

```
Device#clear ipv6 pim traffic
```

# clear ipv6 pimsm-snoop

Clears PIM sparse mode (SM) information.

## Syntax

**clear ipv6 pimsm-snoop** [ **vlan** *vlan-id* ] { **cache** [ *ipv6-address* ] | **stats**}

## Parameters

**vlan***vlan-id*
Specifies clearing information on a specific VLAN.

**cache**
Specifies clearing the PIM SM snooping cache.

*ipv6-address*
Specifies clearing PIM SM snooping-cache information on a specific source or group.

**stats**
Specifies clearing traffic and error counters.

## Modes

Global configuration mode

## Examples

The following example clears PIM SM information from all VLANs.

```
Device(config)# clear ipv6 pimsm-snoop cache
```

The following example clears PIM SM information from a specific VLAN.

```
Device(config)# clear ipv6 pimsm-snoop vlan 10 cache
```

The following example clears PIM SM information from a specific source.

```
Device(config)# clear ipv6 pimsm-snoop cache ff05::100
```

The following example clears traffic and error counters from all VLANs.

```
Device(config)# clear ipv6 pimsm-snoop stats
```

## History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was introduced. |

# clear ipv6 raguard

Resets the drop or permit packet counters for Router Advertisement (RA) guard policies.

## Syntax

**clear ipv6 raguard** { *name* | **all** }

## Parameters

*name*

An ASCII string indicating the name of the RA guard policy of which the packet counters must be cleared.

**all**

Clears the packet counters of all RA guard policies.

## Modes

Global configuration mode

## Usage Guidelines

To clear RA guard packet counters for all RA guard policies, use the **all** keyword. To clear the RA guard packet counters for a specific RA guard policy, specify the *name* of the policy.

## Examples

The following example clears the packet count for an RA guard policy:

```
device(config)# clear ipv6 raguard policy1
```

The following example clears the packet counters for all RA guard policies:

```
device(config)# clear ipv6 raguard all
```

# clear ipv6 rip route

Clears all RIPng routes from the RIPng route table and the IPv6 main route table and resets the routes.

## Syntax

**clear ipv6 rip route**

## Modes

Privileged EXEC mode or any configuration mode.

## Examples

The following example clears all RIPng routes.

```
device# clear ipv6 rip route
```

# clear ipv6 route

Clears IPv6 routes.

## Syntax

**clear ipv6 route** [ **vrf** *vrf-name*] [ *ipv6-prefix***/***prefix-length* ]

## Parameters

**vrf** *vrf-name*

Removes IPv6 routes for the specified VPN Routing/Forwarding (VRF) instance.

*ipv6-prefix***/***prefix-length*

Removes IPv6 routes for the specified IPv6 prefix.

## Modes

Privileged EXEC mode

## Usage Guidelines

The *ipv6-prefix***/***prefix-length* parameter clears routes associated with a particular IPv6 prefix. You must specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

**NOTE**
The L2 and L3 protocols might flap in case the number of L3 routes are more.

## Examples

The following example clears IPv6 routes associated with the prefix 2000:7838::/32.

```
device# clear ipv6 route 2000:7838::/32
```

# clear ipv6 traffic

Clears IPv6 traffic statistics (resets all fields to zero).

## Syntax

**clear ipv6 traffic**

## Modes

Privileged EXEC mode

## Examples

The following example clears the IPv6 traffic statistics.

```
device# clear ipv6 traffic
```

# clear ipv6 tunnel

Clears statistics (resets all fields to zero) for all IPv6 tunnels or for a specific tunnel.

## Syntax

**clear ipv6 tunnel** [ *number* ]

## Parameters

*number*

Specifies the tunnel number.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

You can use the **show ipv6 tunnel** command to verify the results of issuing the **clear ipv6 tunnel**command.

## Examples

The following example clears statistics for tunnel 1.

```
device(config)# clear ipv6 tunnel 1
```

# clear ipv6 tunnel stat

Clears counters for IPv6 tunnel traffic.

## Syntax

**clear ipv6 tunnel stat** *number*

## Parameters

*number*

Specifies the tunnel number.

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be entered in privileged EXEC mode and in any configuration mode. Use the **show ipv6 tunnel traffic** command to verify the results of using this command.

## Examples

The following example clears IPv6 tunnel statistics.

```
device# show ipv6 tunnel traffic

IPSEC Tunnels
 Tunnel Status Packet Received Packet Sent Bytes Received Bytes Sent
 1 up/up 85533517 42780261 360799665060 180384879800
 9 up/up 37985 45674 8079286 9180316
 18 up/up 29805 29531 6688206 6436010

device# clear ipv6 tunnel stat

device# show ipv6 tunnel traffic

IPSEC Tunnels
 Tunnel Status Packet Received Packet Sent Bytes Received Bytes Sent
 1 up/up 0 0 0 0
 9 up/up 0 0 0 0
 18 up/up 0 0 0 0
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# clear ipv6 vrrp statistics

Clears IPv6 Virtual Router Redundancy Protocol (VRRP) statistics.

## Syntax

**clear ipv6 vrrp statistics**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be entered in privileged EXEC mode and in any configuration mode. Entering the command in a configuration mode can be useful if you are configuring IPv6 VRRP options, for example, and want to clear existing VRRP statistics.

## Examples

The following example clears IPv6 VRRP statistics when entered in privileged EXEC mode.

```
device# clear ipv6 vrrp statistics
```

The following example clears IPv6 VRRP statistics when entered in VRID interface configuration mode.

```
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ipv6 vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# clear ipv6 vrrp statistics
```

# clear ipv6 vrrp-extended statistics

Clears IPv6 Virtual Router Redundancy Protocol (VRRP) Extended (VRRP-E) statistics.

## Syntax

**clear ipv6 vrrp-extended statistics**

## Modes

Privileged EXEC mode

## Usage Guidelines

This command can be entered in privileged EXEC mode and in any configuration mode. Entering the command in a configuration mode can be useful if you are configuring IPv6 VRRP-E options, for example, and want to clear existing VRRP-E statistics.

## Examples

The following example clears IPv6 VRRP-E statistics when entered in privileged EXEC mode.

```
device# clear ipv6 vrrp-extended statistics
```

The following example clears IPv6 VRRP-E statistics when entered in VRID interface configuration mode.

```
device(config)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ipv6 2001:DB8::2/24
device(config-if-e1000-1/1/5)# ipv6 vrrp-extended vrid 2
device(config-if-e1000-1/1/5-vrid-2)# clear ipv6 vrrp-extended statistics
```

# clear link-keepalive statistics

Clears the UDLD statistics.

## Syntax

**clear link-keepalive statistics**

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

This command clears the Packets sent, Packets received, and Transitions counters in the **show link-keepalive ethernet** command output.

## Examples

The following example shows how to clear the UDLD port statistics.

```
device# clear link-keepalive statistics
```

# clear link-oam statistics

Clears EFM-OAM statistics from all EFM-OAM-enabled interfaces.

## Syntax

**clear link-oam statistics**

## Modes

Privileged EXEC mode

Global configuration mode

EFM-OAM protocol configuration mode

## Examples

The following example clears EFM-OAM statistics from all EFM-OAM-enabled interfaces.

```
device(config)# clear link-oam statistics
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# clear lldp neighbors

Clears cached LLDP neighbor information.

## Syntax

**clear lldp neighbors** [ **ports** { **all** | **ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ] } ]

## Parameters

**ports**

Clears LLDP neighbor information for ports.

**all**

Clears LLDP neighbor information for all LLDP capable ports.

**ethernet** *stackid/slot/port*

Clears LLDP neighbor information for the specified Ethernet interface.

**to** *stackid/slot/port*

Clears LLDP neighbor information for a range of Ethernet interfaces.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

The device clears cached LLDP neighbor information after a port becomes disabled and the LLDP neighbor information ages out. However, if a port is disabled and then re-enabled before the neighbor information ages out, the device will clear the cached LLDP neighbor information when the port is re-enabled.

## Examples

The following example clears the cached LLDP neighbor information for a specific port.

```
device# clear lldp neighbors ports ethernet 1/1/10
```

The following example clears the cached LLDP neighbor information for all ports.

```
device# clear lldp neighbors ports all
```

# clear lldp statistics

Clears the global and per-port LLDP neighbor statistics on the device.

## Syntax

**clear lldp statistics** [ **all** | **ports** { **all** | **ethernet** *stack-id/slot/port* [ **to** *stack-id/slot/port* | [ **ethernet** *stack-id/slot/port* **to** *stack-id/slot/port* | **ethernet** *stack-id/slot/port* ]... ] } ]

## Parameters

**all**
Clears LLDP neighbor statistics for all LLDP-capable ports.

**ports**
Clears LLDP neighbor statistics for ports.

**all**
Clears LLDP neighbor statistics for all Ethernet interfaces.

**ethernet** *stack-id/slot/port*
Clears LLDP neighbor statistics for the specified Ethernet interface.

**to** *stack-id/slot/port*
Clears LLDP neighbor statistics for a range of Ethernet interfaces.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example clears the LLDP neighbor statistics for all ports.

```
device# clear lldp statistics ports all
```

# clear logging

Clears the log entries from the dynamic buffer, the static buffer, or the local buffer.

## Syntax

**clear logging** [ **dynamic-buffer** | **static-buffer** ]

## Parameters

**dynamic-buffer**
Clears log entries from the dynamic buffer.

**static-buffer**
Clears log entries from the static buffer.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example clears the syslog messages stored in the local buffer.

```
device# clear logging
```

The following example clears the log entries from the dynamic buffer.

```
device# clear logging dynamic-buffer
```

# clear loop-detection

Clears loop detection statistics and enables all Err-Disabled ports.

## Syntax

**clear loop-detection**

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example clears loop detection statistics and enables all Err-Disabled ports.

```
device(config)# clear loop-detection
```

# clear l2protocol dot1q-tunnel counters

Clears all Q-in-Q BPDU tunnel counters.

## Syntax

**clear l2protocol dot1q-tunnel counters** [ *unit / slot / port* | *lag-id* ]

## Parameters

**unit / slot / port**

Specifies the interface from which the tunnel counters are to be cleared.

*lag-id*

Specifies the LAG virtual interface from which the tunnel counters are to be cleared.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

If interfaces are not specified, the Q-in-Q BPDU tunnel counters are cleared for all interfaces.

## Examples

The following example clears all Q-in-Q BPDU tunnel counters.

```
device# clear l2protocol dot1q-tunnel counters
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | The command was introduced. |

# clear mac-address

Clears the MAC addresses.

## Syntax

**clear mac-address** [ *mac-address* | **ethernet** *unit/slot/port* | **lag** *lag-id* | **vlan** *vlan-id* ]

## Parameters

*mac-address*
Clears entries in all VLANs with the specified MAC address.

**ethernet** *unit/slot/port*
Clears the entries on the specified port.

**lag** *lag-id*
Specifies the LAG virtual interface.

**vlan** *vlan-id*
Clears all entries in a VLAN.

## Modes

Privileged EXEC mode

Global configuration mode

Cluster configuration mode

## Examples

The following example shows how to clear the MAC address of a specific VLAN.

```
device# clear mac-address vlan 2
```

The following example shows how to clear all MAC addresses in the system.

```
device# clear mac-address
```

## History

| Release version | Command history |
|---|---|
| 08.0.40 | The **mdup-stats** option was removed as it was supported only on FSX devices. |
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear mac-address cluster

Clears cluster-specific MAC addresses.

## Syntax

**clear mac-address cluster** { *cluster-name* | *cluster-id* } [ **vlan** *vlan-id* ] [ **client** [ *client-name* ] ] [ **local** | **remote** ]

## Parameters

*cluster-name*
: Clears the cluster MAC address entries for the cluster identified by the cluster name.

*cluster-id*
: Clears the cluster MAC address entries for the cluster identified by the cluster ID.

**vlan** *vlan-id*
: Clears the VLAN ID for which you want to clear the MAC address.

**client** *client-name*
: Clears cluster client MAC address entries.

**local**
: Clears the MAC addresses local to the cluster.

**remote**
: Clears the MAC addressses remote to the cluster.

## Modes

Privileged EXEC mode

Global configuration mode

Cluster configuration mode

## Examples

The following example shows how to clear cluster-specific MAC addresses.

```
device# clear mac-address cluster AGG-1 local
```

The following example shows how to clear a MAC address for cluster client for a specific VLAN ID.

```
device# clear mac-address cluster AGG-1 vlan 1 local
```

The following example shows how to clear MAC address for cluster client.

```
device# clear mac-address cluster AGG-1 vlan 2 client 1 local
```

# clear mac-authentication sessions

Clears MAC authentication sessions.

## Syntax

**clear mac-authentication sessions** { **mac-address** *mac-address* | **stack-unit** *id* | **ethernet** *device**/**slot**/**port* }

## Parameters

*mac-address*
Specifies the mac-address from which the MAC authentication sessions are to be cleared.

**stack-unit** *id*
Specifies the stack unit from which the MAC authentication sessions are to be cleared.

**ethernet** *device**/**slot**/**port*
Specifies the interface from which the MAC authentication sessions are to be cleared.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to clear the MAC authentication sessions for either a specified MAC address or an ethernet interface.

## Examples

The following example clears the MAC authentication session for the specified MAC address.

```
device# clear mac-authentication sessions 0000.0034.abd4
```

The following example clears the MAC authentication session sessions on an interface.

```
device# clear mac-authentication sessions ethernet 1/1/1
```

The following example clears the MAC authentication sessions.

```
device# clear mac-authentication sessions
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.70 | The command was modified to include the stack-unit option. |

# clear mac-authentication statistics

Clears MAC authentication statistics

## Syntax

**clear mac-authentication statistics** { **stack-unit** *id* | **ethernet** *device*/*slot*/*port* }

## Parameters

**stack-unit** *id*

Specifies the stack unit from which the MAC authentication statistics are to be cleared.

**ethernet** *device*/*slot*/*port*

Specifies the interface from which the MAC authentication statistics are to be cleared.

## Modes

Privileged EXEC mode

## Examples

The following example clears MAC authentication statistics for stack unit 3.

```
device# clear mac-authentication statictics stack-unit 3
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | The command was modified to include the stack-unit option. |

# clear macsec statistics

Clears the MACsec traffic statistics for the specified interface.

## Syntax

**clear macsec statistics ethernet** *device* **/** *slot* **/** *port*

## Parameters

**ethernet** *device* **/** *slot* **/** *port*

Specifies an interface by device position in stack, slot on the device, and interface on the slot.

## Modes

privileged EXEC mode

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

## Examples

In the following example, MACsec traffic statistics are cleared for interface 1/3/4 (port 4 of slot 3 on the first device in the stack).

```
device# clear macsec statistics ethernet 1/3/4
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# clear management-vrf-stats

Clears the management Virtual Routing and Forwarding (VRF) rejection statistics.

## Syntax

**clear management-vrf-stats**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface management configuration mode

## Usage Guidelines

You can use the **show management-vrf** command to verify the result of issuing the **clear management-vrf-stats** command.

## Examples

The following example clears the management VRF rejection statistics.

```
device(config)# clear management-vrf-stats
```

# clear notification-mac statistics

Clears the MAC-notification statistics, such as the number of trap messages and number of MAC notification events sent.

## Syntax

**clear notification-mac statistics**

## Command Default

The MAC-notification statistics are available on the device.

## Modes

Global configuration

Privileged EXEC

## Usage Guidelines

MAC notification statistics can be viewed using the **show notification-mac** display command.

## Examples

The following example clears the MAC notification statistics:

```
device(config)# clear notification-mac statistics
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# clear openflow

Clears flows from the flow table.

## Syntax

**clear openflow** { **flowid** *flow-id* | **all** }

## Parameters

**flowid** *flow-id*

Clears the given flow ID that you want to delete from the flow table.

**all**

Deletes all flows from the flow table.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

When an OpenFlow rule or all flows in the flow table need to be deleted you can use the **clear openflow** command with the **all** option. To delete a single OpenFlow rule based on a flow-id, use the **clear openflow** command with the **flowid** *flow-id* options.

## Examples

The following example clears the flow with an ID of 6.

```
device# clear openflow flowid 6
```

The following example clears all flows in the flow table.

```
device# clear openflow all
```

## History

| Release | Command History |
|---|---|
| 08.0.20 | This command was introduced. |

# clear port security

Clears port security data.

## Syntax

**clear port security** { **restricted-macs** | **statistics** } { **all** | **ethernet** *stack/slot/port* }

## Parameters

**restricted-macs**
Clears all restricted MAC addresses globally.

**statistics**
Clears violation statistics globally.

**all**
Clears information for all ports.

**ethernet** *stack/slot/port*
Clears information for the specified Ethernet port.

## Modes

Privileged EXEC mode

Global configuration mode

Port security configuration mode

Port security interface configuration mode

## Examples

The following example clears all restricted MAC addresses globally.

```
device# clear port security restricted-macs all
```

The following example clears restricted MAC addresses on a specific port.

```
device# clear port security restricted-macs ethernet 1/1/1
```

The following example clears violation statistics globally.

```
device# clear port security statistics all
```

The following example clears violation statistics on a specific port.

```
device# clear port security statistics ethernet 1/1/1
```

# clear public-key

Clears the authorized client public key from the buffer.

## Syntax

**clear public-key**

## Modes

Privileged EXEC mode

Global configuration mode

## Examples

The following example clears the client public key from the buffer.

```
device# clear public-key
```

# clear pvstplus-protect-statistics

Clears the statistics of the PVST+ Protect feature, configured by means of the **pvstplus-protect** command.

## Syntax

**clear pvstplus-protect-statistics** [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ]

**clear pvstplus-protect-statistics** [ **lag** *lag-id* ]

## Parameters

**ethernet**
Specifies an Ethernet port.

*unit/slot/port*
Number of an Ethernet port. Ranging is allowed by means of the **to** keyword.

**lag** *lag-id*
Specifies the LAG virtual interface.

to
Enables optional ranging.

## Modes

Privileged EXEC mode

## Examples

This example clears the statistics of PVST+ Protect on all Ethernet interfaces, including the number of dropped PVST+ BPDUs.

```
device# clear pvstplus-protect-statistics
```

This example clears the statistics of PVST+ Protect on a single Ethernet interface.

```
device# clear pvstplus-protect-statistics ethernet 1/1/1
```

This example clears the statistics of PVST+ Protect on a range of Ethernet interfaces.

```
device# clear pvstplus-protect-statistics ethernet 1/1/1 to 1/1/4
```

## History

| Release version | Command history |
|---|---|
| 08.0.30mb | This command was introduced. |
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear stack ipc

Clears stack traffic statistics.

## Syntax

**clear stack ipc**

## Command Default

Stack traffic statistics are collected and retained.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **clear stack ipc** command before issuing the **show stack ipc** command. This helps to ensure that the data are the most recent traffic statistics for the stack.

This command must be executed from the active stack controller.

# Examples

The following example clears stack traffic statistics prior to using the **show stack ipc** command to display current stack traffic statistics.

```
device# clear stack ipc
device# show stack ipc
V15, G1, Recv: SkP0:3749372, P1:3756064, MAIL:184291175, sum:191796611, t=457152.2
Message types have callbacks:
1 :Reliable IPC message 2 :Reliable IPC atomic 4 :fragmentation, jumbo
5 :probe by mailbox 6 :rel-mailbox 7 :test ipc
8 :disable keep-alive 9 :register cache 10:ipc dnld stk
11:chassis operation 12:ipc stk boot 13:Rconsole IPC message
14:auth msg 15:ipc erase flash 16:unconfigure
17:ipc stk boot 18:ss set 19:sFlow IPC message
21:SYNC download reques 23:SYNC download 1 spec 28:SYNC client hello
30:SYNC dy chg error 32:active-uprintf 33:test auth msg
34:probe KA 39:unrel-mailbox 40:trunk-probe
Send message types:
[1]=2342639, [4]=44528, [5]=961830, [6]=37146,
[9]=73104634, [11]=137082, [14]=487007, [20]=2304,
[22]=1395, [25]=23, [26]=1901701, [29]=415888,
[34]=1827543, [39]=30451, [40]=289420,
Recv message types:
[1]=2016251, [4]=1352759, [5]=470884, 475144,
[6]=114459, 114572, [9]=367644144, [11]=1785229,
[14]=973285, 974177, [21]=1395, [30]=25,
[34]=912972, 914086, [39]=973492, 973440, [40]=700313,
Statistics:
send pkt num : 34068433, recv pkt num : 191796609,
send msg num : 79756048, recv msg num : 379902767,
send frag pkt num : 22264, recv frag pkt num : 493860,
pkt buf alloc : 34068433,
Reliable-mail send success receive duplic
target ID 1 1 0 0
target MAC 15230 15230 0 0
unrel target ID 7615 0
There is 1 current jumbo IPC session
Possible errors:
*** recv from non-exist unit 2 times: unit 5
```

# History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |

# clear statistics

Clears all counters and statistics.

## Syntax

**clear statistics** [ **dos-attack** | **traffic-policy** *traffic-policy-name* ]

**clear statistics** [ **rate-counters** ] [**ethernet** *unit/slot/port* | **lag** *lag-id* | **management** *number* | **tunnel** [ *number*] | **unit** *number* ]

## Parameters

**dos-attack**
Clears statistics about ICMP and TCP SYN packets dropped because burst thresholds were exceeded.

**traffic-policy** *traffic-policy-name*
Clears traffic policy counters (access list and rate limit counters).

**rate-counters**
Clears the rate counters.

**ethernet** *unit/slot/port*
Clears egress queue statistics (resets the statistics to zero) for all unit/slot/port.

**lag** *lag-id*
Specifies the LAG virtual interface.

**management** *number*
Clears all statistics on a management port.

**tunnel**
Clears all GRE tunnel statistics.

*number*
Clears GRE tunnel statistics for the specified tunnel.

**unit** *number*
Clears a stack unit statistics.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example clears the statistics for a specific Ethernet interface.

```
device(config)# clear statistics ethernet 1/1/1
```

The following example clears the rate counters for a tunnel interface.

```
device(config)# clear statistics rate-counters tunnel 2
```

The following example clears the statistics about ICMP and TCP SYN packets dropped.

```
device(config)# clear statistics dos-attack
```

The following example clears access list and rate limit counters.

```
device(config)# clear statistics traffic-policy counttwo
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear statistics openflow

Clears OpenFlow statistics.

## Syntax

**clear statistics openflow** { **group** | **meter** | **controller** }

## Parameters

**group**
Clears statistics for all groups.

**meter**
Clears statistics for all meters.

**controller**
Clears statistics for all controllers.

## Modes

EXEC and Privileged EXEC mode

Global configuration mode

## Usage Guidelines

This command can be entered in three configuration modes as shown in the examples below.

## Examples

The following example, entered in User EXEC mode, clears statistics for all groups in User EXEC mode.

```
device> clear statistics openflow group
```

The following example, entered in Privileged EXEC mode, clears statistics for all meters in Privileged EXEC mode.

```
device> enable
device# clear statistics openflow meter
```

The following examples, entered in global configuration mode, clears statistics for all controllers.

```
device# configure terminal
device(config) # clear statistics openflow controller
```

## History

| Release | Command History |
|---|---|
| 08.0.20 | This command was introduced. |

# clear stp-protect-statistics

Clears the BPDU drop counters for all ports on the device that have STP Protection enabled.

## Syntax

**clear stp-protect-statistics** [ **ethernet** *unit/slot/port* | **lag** *lag-id* ]

## Parameters

**ethernet** *unit/slot/port*

Specifies the Ethernet interface on which to clear the BPDU drop counters.

**lag** *lag-id*

Specifies the LAG virtual interface.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

For each port that has STP Protection enabled, the ICX device counts and records the number of dropped BPDUs. You can use this command to clear the BPDU drop counters for all ports on the device, or for a specific port on the device.

## Examples

The following example shows how to clear the BPDU drop counters on all ports.

```
device(config)# clear stp-protect-statistics
```

The following example shows how to clear the BPDU drop counter on a specific port.

```
device(config)# clear stp-protect-statistics ethernet 1/1/1
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# clear webauth vlan

Clears the authenticated hosts or the blocked hosts.

## Syntax

**clear webauth vlan** *vlan-id*{ **authenticated-mac** | **block-mac** } [ *mac-address* ]

## Parameters

*vlan-id*

Specifies the VLAN ID.

**authenticated-mac**

Clears authenticated hosts in a Web Authentication VLAN. If a MAC address is specified, then only that host is cleared. If a MAC address is not specified, then all the authenticated hosts are cleared.

**block-mac**

Clears the configured time duration users must wait before the next cycle of Web Authentication attempts is allowed. If a MAC address is specified, then only that host is unblocked. If no MAC address is specified, then all the hosts are unblocked.

*mac-address*

Specifies the MAC address of the host. When used with **authenticated-mac** keyword, this is the dynamically authenticated host MAC address and when used with the **block-mac** keyword, this is the blocked host MAC address.

## Modes

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

Web Authentication configuration mode

## Examples

The following example clears all the authenticated hosts.

```
device# clear webauth vlan 10 authenticated-mac
```

The following example clears the host with MAC address 1111.2222.3333.

```
device# clear webauth vlan 10 authenticated-mac 1111.2222.3333
```

The following example unblocks an authenticated host.

```
device# clear webauth vlan 20 block-mac 1111.2222.3333
```

# clear web-connection

Clears all web management sessions.

## Syntax

**clear web-connection**

## Modes

Privileged EXEC mode

Global configuration mode

## Examples

The following example shows how to clear all the web management sessions.

```
device# clear web-connection
```

# client

Configures cluster clients manually.

## Syntax

**client** *client-name*

**no client** *client-name*

## Command Default

Cluster clients are not configured.

## Parameters

*client-name*

Specifies the name of the client. The client name is an ASCII string and can be up to 64 characters in length.

## Modes

Cluster configuration mode

## Usage Guidelines

Client configuration requires client-name, RBridge ID, and Cluster Client Edge Port (CCEP). The client name can be different on the different cluster devices.

The **no** form of the command removes the manually configured cluster client.

## Examples

The following example shows how to configure the client manually.

```
device(config)# cluster SX 10
device(config-cluster-SX)# client client-2
device(config-cluster-SX-client-2)# rbridge-id 200
device(config-cluster-SX-client-2)# client-interface ethernet 1/2
device(config-cluster-SX-client-2)# deploy
```

# client-auto-detect config

Configures the automatically detected cluster clients into the running configuration and deploys all of the automatically detected clients.

## Syntax

**client-auto-detect config** [ **deploy-all** ]

**no client-auto-detect config** [ **deploy-all** ]

## Command Default

The cluster clients are not automatically detected and deployed.

## Parameters

**deploy-all**

Deploys all automatically detected cluster clients.

## Modes

Cluster configuration mode

## Usage Guidelines

The **no** form of the command removes the configured and deployed automatically detected cluster clients.

## Examples

The following example shows how to configure the automatically detected clients into the running configuration.

```
device(config)# cluster SX 400
device(config-cluster-SX)# client-auto-detect config
```

# client-auto-detect ethernet

Enables cluster client automatic configuration on a specific port or range of ports.

## Syntax

**client-auto-detect ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ... ]

**no client-auto-detect ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ... ]

## Command Default

Cluster client automatic configuration is not enabled on the ports.

## Parameters

**ethernet** *stackid/slot/port*

Specifies the Ethernet port on which you want to enable the cluster client automatic configuration.

**to** *stackid/slot/port*

Specifies the range of ports on which you want to enable the cluster client automatic configuration.

## Modes

Cluster configuration mode

## Usage Guidelines

The **no** form of the command disables the cluster client automatic configuration on the ports.

## Examples

The following example shows how to enable cluster client automatic configuration on an Ethernet port.

```
device(config)# cluster SX 400
device(config-cluster-SX)# client-auto-detect ethernet 1/1/15
```

The following example shows how to enable cluster client automatic configuration on a range of ports.

```
device(config)# cluster SX 400
device(config-cluster-SX)# client-auto-detect ethernet 1/1/15 to 1/1/18
```

# client-auto-detect start

Starts the cluster client automatic configuration.

## Syntax

**client-auto-detect start** [ **config-deploy-all** ]

## Command Default

The client automatic detection process is not enabled.

## Parameters

**config-deploy-all**

Configures and deploys all automatically detected clients.

## Modes

Cluster configuration mode

## Usage Guidelines

Make sure that the network connection and configuration are in place before using this command. Within one minute of the time that each client is discovered, the client is automatically configured and deployed into the running configuration.

Within one minute of configuring this command, the system reports information and errors (if there are mismatches, such as an LACP configuration mismatch). You can fix a mismatch while the process is running.

Use the **config-deploy-all** option as an alternative to the **client-auto-detect config** command. The **client-auto-detect config** command also configures automatically detected clients into the running configuration and deploys all of the automatically detected clients.

## Examples

The following example shows how to start the client automatic configuration process.

```
device(config)# cluster SX 400
device(config-cluster-SX)# client-auto-detect start
```

# client-auto-detect stop

Stops the automatic configuration process of the running cluster client.

## Syntax

**client-auto-detect stop**

## Command Default

The automatic configuration process of the running cluster client is not stopped if the client automatic detection process is enabled using the **client-auto-detect ethernet** command.

## Modes

Cluster configuration mode

## Usage Guidelines

All auto-detected but unconfigured clients will be removed.

## Examples

The following example shows how to stop the automatic configuration process of the running cluster client.

```
device(config)# cluster SX 400
device(config-cluster-SX)# client-auto-detect stop
```

# client-interface

Configures the physical port or static LAG port as the Cluster Client Edge Port (CCEP).

## Syntax

**client-interface** { **ethernet** *unit/slot/port* | **lag** *lag-id* }

**no client-interface** { **ethernet** *unit/slot/port* | **lag** *lag-id* }

## Command Default

A port is not configured as the CCEP.

## Parameters

**ethernet** *unit/slot/port*
Configures the specified Ethernet port as the client CCEP.

**lag** *lag-id*
Configures the specified LAG as the client CCEP.

## Modes

Cluster configuration mode

Cluster client configuration mode

## Usage Guidelines

The **no** form of the command removes the port as the CCEP.

## Examples

The following example shows how to configure a port as the CCEP.

```
device(config)# cluster SX 400
device(config-cluster-SX)# client 1
device(config-cluster-SX-client-1)# client-interface ethernet 1/1/5
device(config-cluster-SX-client-1)# deploy
```

# client-interfaces shutdown

Shuts down all the local client interfaces in the cluster.

## Syntax

**client-interfaces shutdown**

**no client-interfaces shutdown**

## Command Default

Client interfaces are active.

## Modes

Cluster configuration mode

## Usage Guidelines

Use the **client-interfaces shutdown** command when performing a hitless upgrade operation. This command can be used to shut down all the local client interfaces in the cluster, resulting in fail-over of traffic to the peer device.

The **no** form of the command removes the client interface shutdown.

## Examples

The following example shows how to shut down all the client interfaces in the cluster.

```
device(config)# cluster SX 4000
device(config-cluster-SX)# client-interfaces shutdown
```

# client-isolation

Isolates the client from the network when Cluster Communication Protocol (CCP) is not operational.

## Syntax

**client-isolation strict**

**no client-isolation strict**

## Command Default

Client isolation is in loose mode.

## Parameters

**strict**

Specifies the strict isolation mode.

## Modes

Cluster configuration mode

## Usage Guidelines

In strict mode, when the CCP goes down, the interfaces on both the cluster devices are administratively shut down. In strict mode, the client is completely isolated from the network if the CCP is not operational.

In loose mode (default), when the CCP goes down, the peer device performs the master/slave negotiation. After negotiation, the slave shuts down its peer ports, whereas the master peer ports continue to forward the traffic (keep-alive VLAN configured).

MCT cluster devices can operate in two modes. Both peer devices must be configured in the same mode.

**NOTE**
The CLI allows modification of the client isolation mode on MCT cluster devices even when the cluster is deployed. You must create the same isolation mode on both cluster devices.

The **no** form of the command sets client isolation mode back to loose mode.

## Examples

The following example shows how to configure the client isolation strict mode.

```
device(config)# cluster SX 4000
device(config-cluster-SX)# client-isolation strict
```

# client-to-client-reflection

Enables routes from one client to be reflected to other clients by the host device on which it is configured.

## Syntax

**client-to-client-reflection**

**no client-to-client-reflection**

## Command Default

This feature is enabled.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family L2VPN EVPN configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

The host device on which it is configured becomes the route-reflector server.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

This example configures client-to-client reflection on the BGP4 host device.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# client-to-client-reflection
```

This example disables client-to-client reflection on the BGP4+ host device.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# no client-to-client-reflection
```

# clock set

Sets the local clock time and date.

## Syntax

**clock set** *hh:mm:ss mm-dd-yy/yyyy*

## Parameters

*hh:mm:ss*

Specifies the local clock time in hours, minutes, and seconds.

*mm-dd-yy/yyyy*

Specifies the local clock date in month, day, and year format. Year may be specified with two or four numbers.

## Modes

Privileged EXEC mode

## Usage Guidelines

Valid date and time settings range from January 1, 1970 to December 31, 2035.

An active NTP server, if configured, automatically updates and overrides the local clock time.

## Examples

The following example sets the time and date to 31 minutes past 4pm in the afternoon on July 28, 2016, for the local device:

```
device# clock set 16:31:35 07-28-16
```

# clock summer-time

Sets the device clock summer-time and time zone options.

## Syntax

**clock summer-time**

**clock summer-time** [ **zone** { **australia** *australia-time* | **europe** *europe-time* | **gmt** *gmt-time* | **us** *us-time* } **start** *mm-dd-yy hh:mm:ss* **end** *mm-dd-yy hh:mm:ss* ] [ **offset** *offset-value* ]

**no clock summer-time**

**no clock summer-time** [ **zone** { **australia** *australia-time* | **europe** *europe-time* | **gmt** *gmt-time* | **us** *us-time* } **start** *mm-dd-yy hh:mm:ss* **end** *mm-dd-yy hh:mm:ss* ] [ **offset** *offset-value* ]

## Command Default

The default start and end time of day light savings will depend on the longitude of the country. See the Usage Guidelines section for details.

## Parameters

**australia** *australia-time*

Specifies the Australia time zone. The value can be one of the following: cst (UTC+9.5), est (UTC+10), wst (UTC +8).

**europe** *europe-time*

Specifies the Europe time zone. The value can be one of the following: gmt (UTC), bst (UTC+1), ist (UTC+8), wet (UTC), west (UTC+1), cet (UTC+!), cest (UTC+2), eet (UTC+2), eest (UTC+3), msk (UTC+3), msd (UTC+4).

**gmt** *gmt-time*

Specifies the GMT time zone. The value can be one of the following: gmt+00 (United Kingdom), gmt+01 (France, Germany), gmt+02 (Eastern Europe, South Africa), gmt+03, gmt+03:30, gmt+04, gmt+04:30, gmt+05, gmt+05:30 (India), gmt+06, gmt+06:30, gmt+07, gmt+08 (China, Hong Kong, Taiwan), gmt+09 (Japan, Korea), gmt+09:30, gmt+10 (Australia), gmt+10:30, gmt+11, gmt+11:30, gmt+12, gmt-01, gmt-02, gmt-03, gmt-03:30, gmt-04, gmt-05, gmt-06, gmt-07, gmt-08, gmt-08:30, gmt-09, gmt-09:30, gmt-10, gmt-11, gmt-12.

**us** *us-time*

Specifies the US time zone. The value can be one of the following: alaska, aleutian, arizona, central, east-indiana, eastern, hawaii, michigan, mountain, pacific, samoa.

**start** *mm-dd-yy hh:mm:ss*

Specifies the summer-time start date and time for the local clock time in month, day, and year and hours, minutes, and seconds.

**end** *mm-dd-yy hh:mm:ss*

Specifies the summer-time end date and time for the local clock time in month, day, and year and hours, minutes, and seconds.

**offset** *offset-value*

Specifies the summer-time offset, in minutes.

# Modes

Global configuration mode

# Usage Guidelines

The **clock summer-time** command without any parameters sets the default daylight savings time for the corresponding time zone. Use this command with specific parameters if you need to manually configure the local clock summer-time and time zones values. Use the **clock timezone** command to set the device clock time zone with a default daylight savings time.

By default, daylight savings are implemented according to time zone in three sets of dates and times:

- USA—Summer time starts at 2:00am on the second Sunday of March and ends at 2:00am on the first Sunday of November.
- Europe—Summer time starts at 2:00am on the last Sunday of March and ends at 2:00am on the last Sunday of October.
- Rest of the world—Summer time starts at 2:00am on the last Sunday of March and ends at 2:00am on the last Sunday of October, but some countries have different start and end dates depending on the longitude.

When the configured time zone is different from the existing time zone due to a configuration of the time zone using the **clock summer-time** command, a y/n option appears.

The **no** form of this command disables daylight savings.

# Examples

The following example sets the local device clock that resides in the US Central time zone to the US Mountain standard time zone, and you are reminded of this change with a y/n prompt. The daylight savings times are also different than the default for any US time zone.

```
device# configure terminal
Router(config)# clock summer-time zone us mountain start 10-30-16 02:00:00 end 02-27-17 02:00:00 offset
30
You are about to change the timezone config do you want to continue yes or no
(enter 'y' or 'n'): y
```

The following example removes the daylight savings set for the local device clock.

```
device# configure terminal
device(config)# no clock summer-time
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was modified to add subsets of time zones specific to Australia and Europe. |

# clock timezone

Sets the device system clock time zone options using either Greenwich Mean time (GMT) or a specified global region with a subzone that uses Universal Time Coordinated (UTC) plus or minus a number of hours.

## Syntax

**clock timezone** { **australia** *australia-time* | **europe** *europe-time* | **gmt** *gmt-time* | **us** *us-time* }

**no clock timezone** { **australia** *australia-time* | **europe** *europe-time* |**gmt** *gmt-time* | **us** *us-time* }

## Parameters

**australia** *australia-time*

Specifies the Australia time zone. The value can be one of the following: cst (UTC+9.5), est (UTC+10), wst (UTC +8).

**europe** *europe-time*

Specifies the Europe time zone. The value can be one of the following: gmt (UTC), bst (UTC+1), ist (UTC+8), wet (UTC), west (UTC+1), cet (UTC+!), cest (UTC+2), eet (UTC+2), eest (UTC+3), msk (UTC+3), msd (UTC+4).

**gmt** *gmt-time*

Specifies the GMT time zone. The value can be one of the following: gmt+00 (United Kingdom), gmt+01 (France, Germany), gmt+02 (Eastern Europe, South Africa), gmt+03, gmt+03:30, gmt+04, gmt+04:30, gmt+05, gmt+05:30 (India), gmt+06, gmt+06:30, gmt+07, gmt+08 (China, Hong Kong, Taiwan), gmt+09 (Japan, Korea), gmt+09:30, gmt+10 (Australia), gmt+10:30, gmt+11, gmt+11:30, gmt+12, gmt-01, gmt-02, gmt-03, gmt-03:30, gmt-04, gmt-05, gmt-06, gmt-07, gmt-08, gmt-08:30, gmt-09, gmt-09:30, gmt-10, gmt-11, gmt-12.

**us** *us-time*

Specifies the US time zone. The value can be one of the following: alaska, aleutian, arizona, central, east-indiana, eastern, hawaii, michigan, mountain, pacific, samoa.

## Modes

Global configuration mode

## Usage Guidelines

Use this command if you need to manually configure the local clock summer-time and time zones values. Use the **clock timezone** command to set only the clock time zone.

The **no** form of this command resets the default summer-time and zone values.

## Examples

The following example sets the device clock to the Australia western standard time zone.

```
device# configure terminal
device(config)# clock timezone australia wst
```

The following example sets the device clock to the US Mountain time zone.

```
device# configure terminal
device(config)# clock timezone us mountain
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was modified to add subsets of time zones specific to Australia and Europe. |

# cluster

Configures a Multi-Chassis Trunking (MCT) cluster.

## Syntax

**cluster** [ *cluster-name* ] *cluster-id*

**no cluster** [ *cluster-name* ] *cluster-id*

## Command Default

An MCT cluster is not configured.

## Parameters

*cluster-name*

Specifies the cluster name as an ASCII string. The cluster name can be up to 64 characters in length.

*cluster-id*

Specifies the cluster ID. The ID value range can be from 1 through 4095.

## Modes

Global configuration mode

## Usage Guidelines

The *cluster-name* variable is optional; the device auto-generates *cluster-name* as CLUSTER-X when only the cluster ID is specified.

**NOTE**
The *cluster-id* variable must be the same on both cluster devices.

The **no** form of the command removes the MCT cluster configuration.

## Examples

The following example configures an MCT cluster.

```
device(config)# cluster SX 4000
device(config-cluster-SX)# rbridge-id 3
```

# cluster-id

Configures a cluster ID for the route reflector.

## Syntax

**cluster-id** { *num* | *ip-addr* }

**no cluster-id** { *num* | *ip-addr* }

## Command Default

The default cluster ID is the device ID.

## Parameters

*num*

Integer value for cluster ID. Range is from 1 through 65535.

*ip-addr*

IPv4 address in dotted-decimal notation.

## Modes

BGP configuration mode

## Usage Guidelines

When configuring multiple route reflectors in a cluster, use the same cluster ID to avoid loops within the cluster.

The **no** form of the command restores the default.

## Examples

The following example configures a cluster ID for the route reflector.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# cluster-id 1234
```

# compare-routerid

Enables comparison of device IDs, so that the path-comparison algorithm compares the device IDs of neighbors that sent otherwise equal-length paths.

## Syntax

**compare-routerid**

**no compare-routerid**

## Modes

BGP configuration mode

## Examples

The following example configures the device always to compare device IDs.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# compare-routerid
```

# confederation identifier

Configures a BGP confederation identifier.

## Syntax

**confederation identifier** *autonomous-system number*

**no confederation identifier**

## Command Default

No BGP confederation identifier is identified.

## Parameters

*autonomous-system number*

Specifies an autonomous system number (ASN). The configurable range of values is from 1 through 4294967295.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command removes a BGP confederation identifier.

## Examples

The following example specifies that confederation 65220 belongs to autonomous system 100.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 65220
device(config-bgp-router)# confederation identifier 100
```

# confederation peers

Configures subautonomous systems to belong to a single confederation.

## Syntax

**confederation peers** *autonomous-system number* [ *...autonomous-system number* ]

**no confederation peers**

## Command Default

No BGP peers are configured to be members of a BGP confederation.

## Parameters

*autonomous-system number*

Autonomous system (AS) numbers for BGP peers that will belong to the confederation. The configurable range of values is from 1 through 4294967295.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command removes an autonomous system from the confederation.

## Examples

The following example configures autonomous systems 65520, 65521, and 65522 to belong to a single confederation under the identifier 100.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 65020
device(config-bgp-router)# confederation identifier 100
device(config-bgp-router)# confederation peers 65520 65521 65522
```

# console timeout

Configures the idle time for a serial console session.

## Syntax

**console timeout** *time*

**no console timeout** *time*

## Command Default

By default, an ICX device does not time out serial console sessions.

## Parameters

*time*

The time a serial session can remain idle before it is timed out, in minutes. The valid range is from 0 through 240. The default value is 0 (no timeout).

## Modes

Global configuration mode

Stacking configuration mode

## Usage Guidelines

A serial session remains open indefinitely until you close it. You can define how many minutes a serial management session can remain idle before it is timed out.

**NOTE**
You must enable AAA support for console commands, AAA authentication, and EXEC authorization to set the console idle time.

**NOTE**
In RADIUS, the standard attribute Idle-Timeout is used to define the console session timeout value. The attribute Idle-Timeout value is specified in seconds. Within the switch, the idle-Timeout value is truncated to the nearest minute, because the switch configuration is defined in minutes.

You can also configure the console timeout (in minutes) on all stack units (including the Active Controller).

The **no** form of the command removes the timeout settings.

## Examples

The following example shows how to configure the console session timeout as 10 minutes.

```
device(config)# console timeout 10
```

The following example shows how to configure the console timeout on a stack unit.

```
device(config)# stack unit 3
device(config-unit-3)# console timeout 5
```

# copy disk0

Copies the license, running configuration, and startup configuration from disk0 to flash.

## Syntax

**copy disk0** [ **license** | **running-config** | **startup-config** ] *filename*

## Parameters

**license**
Copies the software license from disk0 to flash.

**running-config**
Copies the running configuration from disk0 to flash.

**startup-config**
Copies the startup-configuration from disk0 to flash.

## Modes

Privileged EXEC mode.

## Usage Guidelines

Use the **show files** command to verify if the running configuration and startup configuration are copied to flash correctly. Use the **show license** command to verify if the license is copied correctly.

## Examples

The following example shows copying the license from disk0 to flash.

```
device# copy disk0 license 20140611132829945ICX7450-PREM-LIC-SW.XML unit 1

Copy Software License from disk0 to Flash
```

The following example shows copying the running configuration from disk0 to flash.

```
device# copy disk0 running-config running-config
```

The following example shows copying the log file.

```
device# copy flash disk0 file ./logs/pid-log.txt pid-log-ruckus
Done.
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# copy disk0 flash

Copies configuration data from an external USB disk to flash.

## Syntax

**copy disk0 flash** *file-name* { **bootrom** | **client-certificate** | **client-private-key** | **file** | **fips-bootrom-sig** | **fips-primary-sig** | **fips-secondary-sig** | **local-pri** | **local-sec** | **primary** | **secondary** | **trust-certificate** }

## Parameters

**bootrom**
Specifies a boot ROM image.

**client-certificate**
Specifies a client RSA certificate.

**client-private-key**
Specifies a client RSA private key.

**file** *file-name*
Specifies a file.

**fips-bootrom-sig**
Specifies a boot Rom signature file.

**fips-primary-sig**
Specifies a primary signature file.

**fips-secondary-sig**
Specifies a secondary signature file.

**local-pri**
Specifies a primary code image on the local unit.

**local-sec**
Specifies a ssecondary code image on the local unit.

**primary**
Specifies a primary code image.

**secondary**
Specifies a secondary code image.

**trust-certificate**
Specifies a trust certificate.

## Modes

Privileged EXEC mode.

# Examples

The following example copies a boot ROM image from an external USB disk to flash.

```
device# copy disk0 flash 8.0.60a.bin bootrom
```

# copy flash disk0

Copies the image binary stored in primary or secondary partition of the flash to the external USB flash drive.

## Syntax

**copy flash disk0** { **file | primary | secondary** } *file name*

## Parameters

**file**
Specifies the file to be copied.

**primary**
Specifies the primary partition of the flash where the source file is located.

**secondary**
Specifies the secondary partition of the flash where the source file is located.

## Modes

Privileged EXEC mode.

## Usage Guidelines

Use the **show files disk0** to verify the files copied.

## Examples

The following example shows copying the image binary stored in the primary partition of the flash to the external USB.

```
device# copy flash disk0 primary SWR08030q040.bin
Flash Memory Write (8192 bytes per dot)
................................................................................................
........
Copy Done.
```

The following example shows copying the core files from the flash to disk0.

```
device# copy flash disk0 file ./cores/core_1078-1.gz   core-file

Automatic copy to member units: 1
Flash Memory Write (8192 bytes per dot) ICX7450-48
Switch#.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.........................................................................................
.....................................................................
Copy Done.
```

The following example shows copying the log files from flash to disk0.

```
device# copy flash disk0 file ./logs/pid-log.txt pid-log-ruckus
Done.
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# copy flash flash

Copies the flash image between primary and secondary flash memory or from active controller primary or secondary flash memory to a stack unit.

## Syntax

**copy flash flash** [ **primary** | **secondary** | **unit-id-pri** *unit-num* |**unit-id-sec** *unit-num* ]

## Command Default

N/A

## Parameters

**primary**
Copy secondary flash to primary flash

**secondary**
Copy primary flash to secondary flash

**unit-id-pri** *unit-num*
Copy active primary image to unit ID

**unit-id-sec** *unit-num*
Copy active secondary image to unit ID

*unit-num*
Stack unit ID

## Modes

Privileged EXEC mode

## Usage Guidelines

The command can be used to overcome stack unit image mismatches.

In place of a single unit ID (*unit-num*), the command can accept a list of stack unit IDs, a range of stack unit IDs, or a combination of the two. IDs in a list must be separated by commas. Ranges of IDs are identified by a hyphen. No spaces may be used in lists or ranges.

## Examples

In the following example, active controller primary flash image is copied to stack unit 2.

```
device# copy flash flash unit-id-pri 2
```

In the following example, active controller secondary flash image is copied to a series of stack units (2, 3, and 4) and a range (5-8).

```
device# copy flash flash unit-id-sec 2,3,4,5-8
```

# copy flash scp

Uploads a copy of an OS image file from a FastIron device's primary or secondary flash memory to an SCP server. The syntax for copying an image between two devices under test (DUTs) is different from the syntax for uploading from an ICX device to a Linux or a Windows server.

## Syntax

Syntax for copying an image between two DUTs:

**copy flash scp** { *ipv4-address-* | *ipv4-hostname-* | **ipv6** { *ipv6-address-prefix/prefix-length* | *ipv6-hostname-* } **outgoing-interface** { **ethernet** *stackid/slot/port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename* { **flash:primary** | **secondary** }

Syntax for uploading from an ICX device to a Linux or a Windows server:

**copy flash scp** { *ipv4-address-* | *ipv4-hostname-* | **ipv6** { *ipv6-address-prefix/prefix-length* | *ipv6-hostname-* } **outgoing-interface** { **ethernet** *stackid/slot/port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename* { **primary** | **secondary** }

## Parameters

*ipv4-address-*
: Specifies the IPV4 address of the SCP server.

*ipv4-hostname-*
: Specifies the IP hostname of the SCP server.

**ipv6**
: Specifies the IPV6 address method for SCP file transfer.

*ipv6-address-prefix/prefix-length*
: Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname-*
: Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
: Specifies the interface to be used to reach the remote host.

**ethernet** *stackid/slot/port*
: Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
: Configures a virtual interface (VE) as the outgoing interface.

**public-key**
: Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
: Specifies DSA as the public key authentication.

**rsa**
Specifies RSA as the public key authentication.

*remote-port*
Specifies the remote port number for the TCP connection.

*remote-filename*
Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the filename.

**flash:primary**
Specifies the binary image in primary flash memory. Configure the **flash:primary** keyword when transferring files between DUTs,. See the usage note regarding using this keyword when transferring files between DUTs.

**primary**
Specifies the binary image in primary flash memory.

**secondary**
Specifies the binary image in secondary flash memory.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

**NOTE**
When transferring files between DUTs, you should configure the **flash:primary** keyword instead of the **primary** keyword because the SCP server does not support remote-filename aliases.

# Examples

The following example uploads a copy of an OS image file from the primary flash memory on an ICX device to the SCP server:

```
device# copy flash scp 10.20.1.1 SPS08040-scp.bin primary
device# copy flash scp 10.20.1.1 SPS08040-scp.bin secondary
```

The following example uploads a copy of an OS image file from the primary flash memory on an ICX device to an SCP server with the IP address of 172.26.51.180 :

```
device# copy flash scp 172.26.51.180 filename primary
```

The following example specifies that the SCP connection is established using SSH public key authentication:

```
device# copy flash scp 172.26.51.180 public-key dsa filename primary
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# copy flash tftp

Copies contents on the device flash memory to a TFTP server.

## Syntax

**copy flash tftp** { *ipv4-address* | *ipv6-address* } *file-name* { **file** | **primary** | **secondary** }

## Parameters

*ipv4-address*
Specifies the IPv4 address of the TFTP server.

*ipv6-address*
Specifies the IPv6 address of the TFTP server.

*file-name*
Specifies the name of the file that must be copied from the flash memory to the TFTP server.

**file**
Copies a file from flash memory to the TFTP server.

**primary**
Copies the primary code image to the TFTP server.

**secondary**
Copies the secondary code image to the TFTP server.

## Modes

Privileged EXEC mode

## Examples

The following example copies the primary code image from the device flash to the TFTP server.

```
device# copy flash tftp 192.168.10.1 kxz10100.bin primary
```

# copy https flash

Copies an image from the HTTPS server to the flash memory.

## Syntax

**copy https flash** { *fqdn-name* | *ip-address* } *file-name* { **primary** | **secondary** } [ **port** *port-num* ]

## Parameters

*fqdn-name*
Specifies the fully qualified domain name (FQDN) of the server.

*ip-address*
Specifies an IP address.

*file-name*
Specifies the file name.

**primary**
Specifies the primary partition.

**secondary**
Specifies the secondary partition.

**port** *port-num*
Specifies the TCP server port number. Valid values range from 1 through 65535. If no port number is specified, the default is 443.

## Modes

Privileged EXEC mode

## Usage Guidelines

Be aware when using this command that the flash memory is locked for the entire image download and installation process.

If a unified file image (UFI) is specified, the UFI consists of the application image, the boot code image, and the FI signature in one unified file.

## Examples

The following example copies the “SPR08070b1.bin” image from the HTTPS server to the flash primary partition. IP address 10.1.1.1 is specified and port 876 is specified.

```
device# copy https flash 10.1.1.1 SPR08070b1.bin primary port 876
```

The following example copies the “SPR08070b1.bin” image from the HTTPS server to the flash secondary partition. IP address 10.2.1.1 is specified. Because no port is specified, the default of 443 is used.

```
device# copy https flash 10.2.1.1 SPR08070b1.bin secondary
```

The following example copies a primary UFI image file from the HTTPS server to the flash primary partition. IP address 10.2.1.1 is specified and port 700 is specified. The UFI consists of the application image, the boot code image, and the FI signature in one unified file.

```
device# copy https flash 10.2.1.1 SPR08080b1ufi.bin primary port 700
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# copy https startup-config

Copies a configuration file from the HTTPS server to the startup configuration file.

## Syntax

**copy https startup-config** { *fqdn-name* | *ip-address* } *file-name* [ **port** *port-num* ]

## Parameters

*fqdn-name*
Specifies the fully qualified domain name (FQDN) of the server.

*ip-address*
Specifies an IP address.

*file-name*
Specifies the file name.

**port** *port-num*
Specifies the TCP server port number. Valid values range from 1 through 65535. If no port number is specified, the default is 443.

## Modes

Privileged EXEC mode

## Usage Guidelines

A reboot is required for the new configuration to take effect.

**NOTE**
Use caution when executing this command because the existing startup configuration is overwritten with the new configuration.

## Examples

The following example copies an ICX configuration from the HTTPS server, specifying IP address 10.2.1.1 and the "cfg/backup.cfg" file name. Port number 876 is also specified.

```
device# copy https startup-config 10.2.1.1 cfg/backup.cfg port 876
```

The following example copies an ICX configuration from the HTTPS server, specifying IP address 10.1.1.1 and the "cfg/backup.cfg" file name. Because no port is specified, the default of 443 is used.

```
device# copy https startup-config 10.1.1.1 cfg/backup.cfg
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# copy running-config disk0

Copies the running configuration from internal flash to external USB flash drive.

## Syntax

**copy running-config disk0** {**filename**}

## Parameters

**filename**

Specifies the system's running configuration file.

## Modes

Privileged EXEC.

## Usage Guidelines

Use the **show files** command to verify the running configuration is copied.

## Examples

The following example shows copying the running configuration from the internal flash to the external USB flash drive.

```
device# copy running-config disk0  running-config7750
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# copy running-config https

Uploads a copy of the running configuration file from a FastIron device to an HTTPS server.

## Syntax

**copy running-config https** { *fqdn-name* | *ip-address* } *file-name* [ **port** *port-num* ]

## Parameters

*fqdn-name*
Specifies the fully qualified domain name (FQDN) of the server.

*ip-address*
Specifies an IP address.

*file-name*
Specifies the file name.

**port** *port-num*
Specifies the HTTPS server port. Valid values range from 1 through 65535. If no port number is specified, the default is 443.

## Modes

Privileged EXEC mode

## Examples

The following example uploads a copy of the running configuration file from a device to the HTTPS server, and specifies port 200.

```
device# copy running-config https 10.1.1.1 upload/backup.cfg port 200
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# copy running-config scp

Uploads a copy of the running configuration file from a FastIron device to an SCP server.

## Syntax

**copy running-config scp** { *ipv4-address* | *ipv4-hostname* | **ipv6** { *ipv6-address* | *ipv6-hostname* } **outgoing-interface** { **ethernet** *stackid/slot/port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename*

## Parameters

*ipv4-address*
: Specifies the IPV4 address of the SCP server.

*ipv4-hostname*
: Specifies the IP hostname of the SCP server.

**ipv6**
: Specifies the IPV6 address method for SCP file transfer.

*ipv6-address*
: Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname*
: Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
: Specifies the interface to be used to reach the remote host.

**ethernet** *stackid/slot/port*
: Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
: Configures a virtual interface (VE) as the outgoing interface.

**public-key**
: Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
: Specifies DSA as the public key authentication.

**rsa**
: Specifies RSA as the public key authentication.

*remote-port*
: Specifies the remote port number for the TCP connection.

*remote-filename*
: Specifies the name of the file in the SCP server that is going to be uploaded. You can specify up to 127 characters for the filename.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

# Examples

The following example uploads a copy of the running configuration file from a FastIron device to a 172.26.51.180 SCP server:

```
device# copy running-config scp 172.26.51.180 runConfig
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# copy running-config tftp

Uploads a copy of the running configuration file from a Layer 2 or Layer 3 switch to a Trivial File Transfer Protocol (TFTP) server.

## Syntax

**copy running-config tftp** *tftp-ip-addr file-name*

## Parameters

*tftp-ip-addr*
The IPv4 or IPv6 address of the TFTP server.

*file-name*
Specifies the file name.

## Modes

Privileged EXEC mode

## Examples

The following example uploads a copy of the running configuration file to a TFTP server.

```
device# copy running-config tftp 192.168.14.26 copyrun
```

# copy scp flash

Downloads a copy of the OS image file from an SCP server to primary or secondary flash memory, or downloads a copy of the boot file or signature file to the device.

## Syntax

Syntax for copying an image between two DUTs:

**copy scp flash** { *ipv4-address* | *ipv4-hostname* | **ipv6** { *ipv6-address* | *ipv6-hostname* } **outgoing-interface** { **ethernet** *stackid***/***slot***/***port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename* { { **flash:primary** | **secondary** } | **bootrom** | { **fips-bootrom-sig** | **fips-primary-sig** | **fips-secondary-sig** | **fips-ufi-primary-sig** | **fips-ufi-secondary-sig** } }

Syntax for downloading to a DUT from a Linux or a Windows server:

**copy scp flash** { *ipv4-address* | *ipv4-hostname* | **ipv6** { *ipv6-address* | *ipv6-hostname-* } **outgoing-interface** { **ethernet** *stackid***/***slot***/***port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename* { { **primary** | **secondary** } | **bootrom** | { **fips-bootrom-sig** | **fips-primary-sig** | **fips-secondary-sig** | **fips-ufi-primary-sig** | **fips-ufi-secondary-sig** } }

## Parameters

*ipv4-address*
: Specifies the IPV4 address of the SCP server.

*ipv4-hostname*
: Specifies the IP host name of the SCP server.

**ipv6**
: Specifies the IPV6 address method for SCP file transfer.

*ipv6-address*
: Specifies the IPV6 address of the SCP server.

*ipv6-hostname*
: Specifies the IPv6 host name of the SCP server.

**outgoing-interface**
: Specifies the interface to be used to reach the remote host.

**ethernet** *stackid***/***slot***/***port*
: Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
: Configures a virtual interface (VE) as the outgoing interface.

**public-key**
: Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
: Specifies DSA as the public key authentication.

**rsa**
Specifies RSA as the public key authentication.

*remote-port*
Specifies the remote port number for the TCP connection.

*remote-filename*
Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the file name.

**flash:primary**
Specifies the binary image in primary flash memory.

**primary**
Specifies that the file being copied is a primary image file or a primary UFI image file.

**secondary**
Specifies that the file being copied is a secondary image file or a secondary UFI image file.

**bootrom**
Specifies the boot ROM file image in the SCP server.

**fips-bootrom-sig**
Specifies the FIPS boot ROM signature file name in the SCP server.

**fips-primary-sig**
Specifies the FIPS primary signature file name in the SCP server.

**fips-secondary-sig**
Specifies the FIPS secondary signature file name in the SCP server.

**fips-ufi-primary-sig**
Specifies that the file being copied is a FIPS primary unified file image (UFI) signature file.

**fips- ufi-secondary-sig**
Specifies that the file being copied is a FIPS secondary UFI signature file.

# Modes

Privileged EXEC mode

# Usage Guidelines

When you configure this command, you are prompted for the username and password.

When transferring files between devices under test (DUTs), you should configure the **flash:primary** keyword instead of the **primary** keyword because the SCP server does not support *remote-filename* aliases.

The syntax for copying an image between two devices under test (DUTs) is different from the syntax for downloading from a DUT to a Linux or a Windows server.

If using the **fips-ufi-primary-sig** or **fips-ufi-secondary-sig** keyword, the file name must be in ASCII text and must contain the .sig extension. The unified file image (UFI) consists of the application image, the boot code image, and the FI signature in one unified file.

The .sig file must be used when FIPS is enabled to validate the unified image with the corresponding signature file.

UFI image download is supported using TFTP, USB, and SCP only.

# Examples

The following example copies an image from an SCP server to an ICX device.

```
device# copy scp flash 10.20.1.1 SPR08030.bin primary
device# copy scp flash 10.20.1.1 SPR08030.bin secondary
```

The following example downloads a copy of the signature file from the SCP server to an ICX device.

```
device# copy scp flash 172.26.51.180 /tftpboot/ICX7450.sig fips-primary-sig
```

The following example copies a FIPS UFI signature file from the SCP server to the primary flash memory.

```
device# copy scp flash 10.37.2.40 signature_ufi.sig fips-ufi-primary-sig
```

The following example copies a FIPS UFI signature file from the SCP server to the secondary flash memory.

```
device# copy scp flash 10.37.2.40 signature_ufi.sig fips-ufi-secondary-sig
```

The following example copies a UFI file from the SCP server to the primary flash memory.

```
device# copy scp flash 10.2.3.4 SPR08080b1ufi.bin primary
```

The following example copies a UFI file from the SCP server to the secondary flash memory.

```
device# copy scp flash 10.2.3.4 SPR08080b1ufi.bin secondary
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.40 | The **icx6450** and **icx6610** options were removed because they are supported only on ICX 6450 and ICX 6610 devices respectively. |
| 08.0.80 | This command was modified to add the **fips-ufi-primary-sig** and **fips-ufi-secondary-sig** keywords, and to download a UFI file to flash memory. |

# copy scp license

Downloads a copy of the license file from an SCP server to the FastIron device.

## Syntax

**copy scp license** { *ipv4-address-* | *ipv4-hostname-* | **ipv6** { *ipv6-address-* | *ipv6-hostname-* } **outgoing-interface** { **ethernet** *stackid**/**slot**/**port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename* [ **unit** *unit-id* ]

## Parameters

*ipv4-address-*
: Specifies the IPV4 address of the SCP server, using 8-bit values in dotted decimal notation.

*ipv4-hostname-*
: Specifies the IP hostname of the SCP server.

**ipv6**
: Specifies the IPV6 address method for SCP file transfer.

*ipv6-address-prefix/prefix-length*
: Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname-*
: Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
: Specifies the interface to be used to reach the remote host.

**ethernet** *stackid**/**slot**/**port*
: Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
: Configures a virtual interface (VE) as the outgoing interface.

**public-key**
: Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
: Specifies DSA as the public key authentication.

**rsa**
: Specifies RSA as the public key authentication.

*remote-port*
: Specifies the local port number for the TCP connection.

*remote-filename*
: Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the filename.

**unit** *unit-id*

Specifies the unit ID of the device in the stack. If two or more pizza-box devices are connected and acting as a single device, a single management ID is assigned to the stack.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

# Examples

The following example downloads a copy of the license file from an SCP server to a FastIron device:

```
Device# copy scp license 172.26.21.180 /tftpboot/abc.xml unit 1
Device#
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# copy scp running-config

Downloads a copy of the running configuration file from an SCP server to a FastIron device.

## Syntax

**copy scp running-config** { *ipv4-address* | *ipv4-hostname* | **ipv6** { *ipv6-address* | *ipv6-hostname* } [ **outgoing-interface** { **ethernet** *stackid*/*slot*/*port* | **ve** *ve-number* } ] } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename* **overwrite**

## Parameters

*ipv4-address*
: Specifies the IPV4 address of the SCP server.

*ipv4-hostname*
: Specifies the IP hostname of the SCP server.

**ipv6**
: Specifies the IPV6 address method for SCP file transfer.

*ipv6-address-prefix*
: Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname*
: Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
: Specifies the interface to be used to reach the remote host.

**ethernet** *stackid*/*slot*/*port*
: Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
: Configures a virtual interface (VE) as the outgoing interface.

**public-key**
: Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
: Specifies DSA as the public key authentication.

**rsa**
: Specifies RSA as the public key authentication.

*remote-port*
: Specifies the remote port number for the TCP connection.

*remote-filename*
: Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the filename.

**overwrite**

Specifies that the FastIron device should overwrite the current configuration file with the copied file. If you do not specify the **overwrite** keyword, the device copies the downloaded file into the current running or startup configuration but does not overwrite the current configuration.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

# Examples

The following example downloads a copy of the running configuration file from an SCP server to a FastIron device:

```
device# copy scp running-config 172.26.51.180 abc.cfg
```

The following example downloads a copy of the running configuration file from an SCP server to a FastIron device and overwrite the current configuration file with the copied file:

```
device# copy scp running-config 172.26.51.180 abc.cfg overwrite
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# copy scp startup-config

Downloads a copy of the startup configuration file from an SCP server to a FastIron device.

## Syntax

**copy scp startup-config** { *ipv4-address* | *ipv4-hostname* | **ipv6** { *ipv6-address* | *ipv6-hostname* } **outgoing-interface** { **ethernet** *stackid***/***slot***/***port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename*

## Parameters

*ipv4-address*
Specifies the IPV4 address of the SCP server, using 8-bit values in dotted decimal notation.

*ipv4-hostname*
Specifies the IP hostname of the SCP server.

**ipv6**
Specifies the IPV6 address method for SCP file transfer.

*ipv6-address*
Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname*
Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
Specifies the interface to be used to reach the remote host.

**ethernet** *stackid***/***slot***/***port*
Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
Configures a virtual interface (VE) as the outgoing interface.

**public-key**
Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
Specifies DSA as the public key authentication.

**rsa**
Specifies RSA as the public key authentication.

*remote-port*
Specifies the remote port number for the TCP connection.

*remote-filename*
Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the filename.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

# Examples

The following example downloads a copy of the startup configuration file from an SCP server to a FastIron device:

```
device# copy scp startup-config 172.26.51.180 abc.cfg
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# copy startup-config disk0

Copies the configuration file present on the external USB to the systems startup configuration file.

## Syntax

**copy startup-config disk0** { **filename** }

## Parameters

**filename**

The system's startup configuration file.

## Modes

Privileged EXEC.

## Usage Guidelines

Use the **show files** command to verify the startup configuration is copied.

## Examples

The following example shows copying the configuration file from the external USB to the system's startup configuration file.

```
device# copy startup-config disk0 startup-config7750
SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE UNIT(8192 bytes per dot)...

Done.

Copy Done.
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# copy startup-config https

Uploads a copy of the startup configuration file from a FastIron device to an HTTPS server.

## Syntax

**copy startup-config https** { *fqdn-name* | *ip-address* } *file-name* [ **port** *port-num* ]

## Parameters

*fqdn-name*
Specifies the fully qualified domain name (FQDN) of the server.

*ip-address*
Specifies an IP address.

*file-name*
Specifies the file name.

**port** *port-num*
Specifies the HTTPS server port. Valid values range from 1 through 65535. If no port number is specified, the default is 443.

## Modes

Privileged EXEC mode

## Usage Guidelines

If no startup configuration is present on the flash, an error message appears and HTTPS upload does not occur.

## Examples

The following example uploads a copy of the startup configuration file from a device to the HTTPS server. Because no port is specified, the default of 443 is used.

```
device# copy startup-config https 10.1.1.1 backup/icx.cfg
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# copy startup-config scp

Uploads a copy of the startup configuration file from a FastIron device to an SCP server.

## Syntax

**copy startup-config scp** { *ipv4-address-* | *ipv4-hostname-* | **ipv6** { *ipv6-address-* | *ipv6-hostname-* } **outgoing-interface** { **ethernet** *stackid**/**slot**/**port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename*

## Parameters

*ipv4-address-*
: Specifies the IPV4 address of the SCP server, using 8-bit values in dotted decimal notation.

*ipv4-hostname-*
: Specifies the IP hostname of the SCP server.

**ipv6**
: Specifies the IPV6 address method for SCP file transfer.

*ipv6-address-prefix/prefix-length*
: Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname-*
: Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
: Specifies the interface to be used to reach the remote host.

**ethernet** *stackid**/**slot**/**port*
: Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
: Configures a virtual interface (VE) as the outgoing interface.

**public-key**
: Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
: Specifies DSA as the public key authentication.

**rsa**
: Specifies RSA as the public key authentication.

*remote-port*
: Specifies the remote port number for the TCP connection.

*remote-filename*
: Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the filename.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

# Examples

The following example uploads a copy of the startup configuration file from a FastIron device to a to a 172.26.51.180 SCP server:

```
device# copy startup-config scp 172.26.51.180 my_startup_file
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# copy startup-config tftp

Uploads a copy of the startup configuration file from a Layer 2 or Layer 3 switch to a TFTP server.

## Syntax

**copy startup-config tftp** *tftp-ip-addr file-name*

## Parameters

*tftp-ip-addr*

The IPv4 or IPv6 address of the TFTP server.

*file-name*

Specifies the file name.

## Modes

Privileged EXEC mode

## Examples

The following example uploads a copy of the startup configuration file to a TFTP server.

```
device# copy startup-config tftp 2001:db8::12:14 file4
```

# copy tftp flash

Downloads files from a TFTP server to the flash memory of a device.

## Syntax

**copy tftp flash** { *ipv4-address* | *ipv6-address* } *file-name* { **bootrom** | **client-certificate** | **client-private-key** | **fips-bootrom-sig** | **fips-primary-sig** | **fips-secondary-sig** | **fips-ufi-primary-sig** | **fips-ufi-secondary-sig** | **primary** | **secondary** | **trust-certificate** }

## Parameters

*ipv4-address*
Specifies the IPv4 address of the TFTP server from where the file must be copied to the device.

*ipv6-address*
Specifies the IPv6 address of the TFTP server from where the file must be copied to the device.

*file-name*
Specifies the name of the file that must be copied from the TFTP server.

**bootrom**
Specifies that the file being copied is a boot ROM image.

**client-certificate**
Specifies that the file being copied is an RSA client certificate file.

**client-private-key**
Specifies that the file being copied is a client RSA private key file.

**fips-bootrom-sig**
Specifies that the file being copied is a FIPS boot ROM signature file.

**fips-primary-sig**
Specifies that the file being copied is a FIPS primary signature file.

**fips-secondary-sig**
Specifies that the file being copied is a FIPS secondary signature file.

**fips-ufi-primary-sig**
Specifies that the file being copied is a FIPS primary unified file image (UFI) signature file.

**fips-ufi-secondary-sig**
Specifies that the file being copied is a FIPS secondary UFI signature file.

**primary**
Specifies that the file being copied is a primary image file or a primary UFI file.

**secondary**
Specifies that the file being copied is a secondary image file or a secondary UFI file.

**trust-certificate**
Specifies that the file being copied is an SSL trust certificate.

# Modes

Privileged EXEC mode

# Usage Guidelines

If the device has 8 MB of flash memory, you must delete the primary and secondary images.

Ruckus recommends that you use the **copy tftp flash** command to copy the boot code to the device during a maintenance window. Attempting to do so during normal networking operations may cause disruption to the network.

If using the **fips-ufi-primary-sig** or **fips-ufi-secondary-sig** keyword, the file name must be in ASCII text and must contain the .sig extension.

The unified file image (UFI) consists of the application image, the boot code image, and the signature in one unified file.

The .sig file must be used when FIPS is enabled to validate the unified image with the corresponding signature file.

UFI image download is supported using TFTP, USB, and SCP only.

# Examples

The following example copies a boot ROM image from the TFTP server.

```
device# copy tftp flash 192.168.10.1 spz10105.bin bootrom
```

The following example copies an image to the primary flash memory.

```
device# copy tftp flash 192.168.10.1 SPS08030.bin primary
```

The following example copies an image to the secondary flash memory.

```
device# copy tftp flash 10.2.3.4 SPR08080b1ufi.bin secondary
```

The following example copies a FIPS UFI signature file from the TFTP server to the primary flash memory.

```
device# copy tftp flash 10.37.2.40 signature_ufi.sig fips-ufi-primary-sig
```

The following example copies a FIPS UFI signature file from the TFTP server to the secondary flash memory.

```
device# copy tftp flash 10.37.2.40 signature_ufi.sig fips-ufi-secondary-sig
```

The following example copies a UFI image file from the TFTP server to the primary flash memory.

```
device# copy tftp flash 10.2.3.4 SPR08080b1ufi.bin primary
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was modified to add the **fips-ufi-primary-sig** and **fips-ufi-secondary-sig** keywords were added, and to download a UFI file to flash memory. |

# copy tftp license

Copies the license file from the TFTP server to the license database of the ICX device.

## Syntax

**copy tftp license** { *ip_address* | *ipv6_address* } *license_filename_on_host* **unit** *unit_id*

## Command Default

By default, the command is not enabled.

## Parameters

*ip_address*
Specifies the address of the IPv4 TFTP server.

*ipv6_address*
Specifies the address of the IPv6 TFTP server.

*license_filename_on_host*
Specifies the filename of the license file.

**unit** *unit_id*
Indicates the specific unit you want to copy the license file to. The *unit_id* can be from 1 through 12 on ICX devices.

## Modes

Privileged EXEC level.

## Usage Guidelines

To remove a license file, use the **license delete** command.

The **unit** *unit_id* parameter is used only on ICX 7450 devices when copying a license file to a specific unit id.

If you attempt to download the same license twice on the device, the following error message is displayed on the console.

```
Can't add the license string - 93 (DUPLICATE_LICENSE)
```

## Examples

The following example copies a license file from the active unit to all other member units in the system.

```
device# copy tftp license 10.120.54.185 ICX7450_LIC_PERP.xml unit 2
```

# copy tftp running-config

Downloads configuration information from a TFTP server into the device's running configuration.

## Syntax

**copy tftp running-config** *ip-addr file-name* [ **overwrite** ]

## Parameters

*ip-addr*

The IPv4 or IPv6 address of the TFTP server.

*file-name*

Specifies the file name on the TFTP server.

**overwrite**

Overwrites the current running configuration.

## Modes

Privileged EXEC mode

## Examples

The following example downloads configuration information into the running configuration.

```
device# copy tftp running-config 2001:db8::12:13 runningfile
```

# copy tftp startup-config

Downloads a copy of the startup configuration file from a TFTP server to a Layer 2 or Layer 3 switch.

## Syntax

**copy tftp startup-config** *tftp-ip-addr filename*

## Parameters

*tftp-ip-addr*

The IPv4 or IPv6 address of the TFTP server.

*file-name*

Specifies the file name of teh TFTP server.

## Modes

Privileged EXEC mode

## Examples

The following example downloads a copy of the startup configuration file from the specified TFTP server.

```
device# copy tftp startup-config 2001:db8::12:13 configfile
```

# copy tftp system-manifest

Simplifies the software upgrade process into a single command.

## Syntax

**copy tftp system-manifest** { *ipv4-address* | *ipv6-address* } *file-name* { **all-images-primary** | **all-images-secondary** | **primary** | **secondary** } [ **router-image** | **switch-image** ]

## Parameters

*ipv4-address*
Specifies the IPv4 address of the TFTP server from where the file must be copied to the device.

*ipv6-address*
Specifies the IPv6 address of the TFTP server from where the file must be copied to the device.

*file-name*
Specifies the name of the file that must be copied from the TFTP server.

**all-images-primary**
Specifies that the file being copied is a primary signature file.

**all-images-secondary**
Specifies that the file being copied is a secondary signature file.

**primary**
Specifies that the file being copied is a primary image file.

**secondary**
Specifies that the file being copied is a secondary image file.

**router-image**
Specifies that the file being copied is a router image.

**switch-image**
Specifies that the file being copied is a switch image.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **all-images-primary** and **all-images-secondary** options upgrade the boot and application images in a single step.

This command only accepts manifest files with a .txt extension. Before starting any download, the file is checked for the correct keywords and extracts the image name and location.

The manifest file consists of images of both router and switch type. Commands in the file check if the system is running a router or a switch image and then installs the appropriate images.

In an 802.1br SPX system, the command can be entered from the master active controller of the control bridge (CB) stack or from a standalone ICX 7750 or ICX 7650 acting as the CB. The manifest file download installs the correct image on all the CB units in the CB stack first, then installs the correct router image on all PE units in the SPX system.

After the relevant images have been installed on the system, the user is notified that the upgrade is complete and is prompted to reload the system for the new images to take effect.

# Examples

The following example downloads all boot and application images for FastIron 8.0.40 from the specified TFTP server location.

```
device# copy tftp system-manifest 10.70.42.172 stage/FI08040_Manifest.txt
all-images-secondary
You are about to download boot image and boot signature image as well, ARE YOU SURE?
(enter 'y' or 'n'): y
device#Flash Memory Write (8192 bytes per dot)
DOWNLOADING MANIFEST FILE Done.
device#Flash Memory Write (8192 bytes per dot)
Automatic copy to member units: 3
COPY ICX7750 SIGNATURE TFTP to Flash Done
device#Load to buffer (8192 bytes per dot)
Automatic copy to member units: 3
...
SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE UNIT(8192 bytes per
dot)...
...
Copy ICX7750 from TFTP to Flash Done.
device#Flash Memory Write (8192 bytes per dot)
Automatic copy to member units: 3
...
DOWNLOAD OF ICX7750 BOOT SIGNATURE Done.
device#Load to buffer (8192 bytes per dot)
Automatic copy to member units: 3
...
SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE UNIT(8192 bytes per
dot)...
...
ICX7750 Boot IMAGE COPY IS DONE
device#Load to buffer (8192 bytes per dot)
Automatic copy to member units: 17 18
...
PLEASE WAIT. MEMBERS SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE
UNIT... Done.
device#Load to buffer (8192 bytes per dot)
Automatic copy to member units: 17 18
...
PLEASE WAIT. MEMBERS SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE
UNIT...
Manifest image download is complete, please reload the system
```

The following example copies the binary image for the FastIron 8.0.40 manifest file to secondary flash from the specified TFTP server location.

```
device# copy tftp system-manifest 10.70.42.172 stage/FI08040_Manifest.txt
s secondary
device# Flash Memory Write (8192 bytes per dot) .....
DOWNLOADING MANIFEST FILE Done.
device#Flash Memory Write (8192 bytes per dot)
Automatic copy to member units: 3
...
COPY ICX7750 SIGNATURE TFTP to Flash Done
device# Load to buffer (8192 bytes per dot)
Automatic copy to member units: 3
...
SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE UNIT(8192 bytes per
dot)...
...
Copy ICX7750 from TFTP to Flash Done.
device#Load to buffer (8192 bytes per dot)
Automatic copy to member units: 17 18
...
PLEASE WAIT. MEMBERS SYNCING IMAGE TO FLASH. DO NOT SWITCH OVER OR POWER DOWN THE
UNIT...
Copy ICX7450 from TFTP to Flash Done.
Manifest file upgrade done, please reload the system
```

**NOTE**

In these examples the device is an ICX 7750.

# History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |
| 08.0.80 | This command was modified. The **router-image** and **switch-image** keywords were added. |

# cpu-limit

Configures a rate limit to control the number of CPU address messages.

## Syntax

**cpu-limit addr-msgs** *number*

**no cpu-limit addr-msgs** *number*

## Parameters

**addr-msgs** *number*

The number of address messages the CPU handles per second. The range for this rate limit is from 200 through 50,000 address messages per second.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The address learning rate limit applies to each packet processor, which means that for a system with two packet processors, each processor can send address messages to the CPU at the established rate limit.

**NOTE**
Actual rates of address messages in hardware may have a variance of +200 or -100.

The **no** form of the command clears the rate limit for the address messages.

## Examples

The following example sets the CPU address rate limit to 200.

```
device(config)# cpu-limit addr-msgs 200
```

# critical-vlan

Specifies the VLAN into which the client should be placed when the RADIUS server times out while authenticating or re-authenticating users.

## Syntax

**critical-vlan** *vlan-id*

**no critical-vlan** *vlan-id*

## Command Default

The client is not part of the critical VLAN.

## Parameters

*vlan-id*

Specifies the VLAN ID of the specific critical VLAN.

## Modes

Authentication configuration mode

## Usage Guidelines

When critical VLAN is configured and the authentication time out action is specified as critical VLAN under the port using the **authentication timeout-action critical-vlan** command at the interface level and if RADIUS timeout happens, the client is moved to the critical VLAN and any access policies applied to the critical VLAN is applied to the client.

The VLAN which is configured as a critical VLAN must be a valid VLAN configured on the device.

The **no** form of the command disables the critical VLAN by removing the client from the VLAN.

## Examples

The following example configures VLAN 20 as critical VLAN.

```
device(config)# authentication
device(config-authen)# critical-vlan 20
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# crl-query (PKI)

Sets the Certificate Revocation List (CRL) query URL.

## Syntax

**crl-query** { *url*}

**no crl-query**

## Command Default

No CRL URL is specified by default.

## Parameters

*url*

URL of the CRL Distribution Point.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the URL configuration.

The CRL Distribution Point (CDP) is used to retrieve a CA's latest CRL, usually an LDAP server or HTTP (web) server. The CDP is normally expressed as an ldap://host/dir or http://host/path URL.

## Examples

The following example configures the HTTP address shown as the URL to be queried for the latest Certificate Revocation List.

```
device# configure terminal
device(config)# pki trustpoint trust1

device(config-pki-trustpoint-trust1)# revocation-check crl
device(config-pki-trustpoint-trust1)# crl-query http://FI-PKI02.englab.ruckus.com/CertEnroll/englab-FI-
PKI02-CA.crl
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# crl-update-time (PKI)

Sets the frequency of CRL updates.

## Syntax

**crl-update-time** { *hours* }

**no crl-update-time**

## Command Default

## Parameters

*hours*

Defines the number of hours between CRL updates. The valid range is 1 through 1000 hours.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

A periodic CRL timer runs and after every expiry, it dumps the entire list of revocation information. The revocation check is done when the CRL information is downloaded for the first time. When the subsequent timer expires, the revocation check is not done unless the tunnels are forced to re-negotiated.

## Examples

The following example sets the CRL update frequency to one hour.

```
device(config)#pki trustpoint trust1
device(config-pki-trustpoint-trust1)# revocation-check crl
device(config-pki-trustpoint-trust1)# crl-query http://FI-PKI02.englab.ruckus.com/CertEnroll/englab-FI-
PKI02-CA.crl
device(config-pki-trustpoint-trust1)# crl-update-time 1
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# crypto key client generate

Generates the crypto client key to enable SSH2.

## Syntax

**crypto key client generate** { **dsa** | **rsa** [ **modulus** *key-size* ] }

## Command Default

The crypto client key is not generated and SSH2 is not enabled.

## Parameters

**dsa**

Generates a DSA client key pair.

**rsa**

Generates an RSA client key pair.

**modulus** *key-size*

Specifies the modulus size of the RSA key pair, in bits. The valid values for the modulus size are 1024 or 2048. The default value is 1024.

## Modes

Global configuration mode

## Usage Guidelines

The **dsa** keyword is optional. If you do not enter the dsa keyword, the crypto key generate command generates a DSA key pair by default.

To use the SSH client for public key authentication, you must generate SSH client authentication keys and export the public key to the SSH servers to which you want to connect.

To disable SSH, you delete all of the client keys from the device. When a client key is deleted, it is deleted from the flash memory of all management modules.

An RSA key with modulus 2048 must be used in FIPS or Common Criteria mode.

## Examples

The following example shows how to generate the DSA client key pair.

```
device(config)# crypto key client generate dsa
```

The following example shows how to generate the RSA key pair.

```
device(config)# crypto key client generate rsa modulus 2048
```

# crypto key client zeroize

Deletes the crypto client key pair from the flash memory.

## Syntax

**crypto key client zeroize** { **dsa** | **rsa** }

## Parameters

**dsa**

Deletes a DSA client key pair.

**rsa**

Deletes an RSA client key pair.

## Modes

Global configuration mode

## Usage Guidelines

To disable SSH, you delete all of the client keys from the device. When a client key is deleted, it is deleted from the flash memory of all management modules.

## Examples

The following example shows how to delete the DSA client key pair.

```
device(config)# crypto key client zeroize dsa
```

The following example shows how to delete the RSA client key pair.

```
device(config)# crypto key client zeroize rsa
```

The following example shows how to delete DSA and RSA client key pairs from flash memory.

```
device(config)# crypto key client zeroize
```

# crypto key generate

Generates the crypto key to enable SSH.

## Syntax

### *Syntax*

**crypto key generate** { **ec** *label_name* [ **size** { **256** | **384** } ] }

**crypto key generate** [ **dsa** | **rsa** [ **modulus** *key-size* ] ]

## Command Default

A crypto key is not generated and SSH is not enabled.

## Parameters

**dsa**

Generates the DSA host key pair.

**rsa**

Generates the RSA host key pair.

**modulus** *key-size*

Specifies the modulus size of the RSA key pair, in bits. The valid values for the modulus size are 1024 or 2048. The default value is 1024.

## Modes

Global configuration mode

## Usage Guidelines

The **dsa** keyword is optional. If you do not enter the dsa keyword, the crypto key generate command generates a DSA key pair by default.

To enable SSH, you generate a DSA or RSA host key on the device. The SSH server on the ICX device uses this host DSA or RSA key, along with a dynamically generated server DSA or RSA key pair, to negotiate a session key and encryption method with the client trying to connect to it. While the SSH listener exists at all times, sessions cannot be started from clients until a host key is generated. After a host key is generated, clients can start sessions. When a host key is generated, it is saved to the flash memory of all management modules. The time to initially generate SSH keys varies depending on the configuration, and can be from a under a minute to several minutes.

To disable SSH, you delete all of the host keys from the device. When a host key is deleted, it is deleted from the flash memory of all management modules.

An RSA key with modulus 2048 must be used in FIPS or Common Criteria mode.

# Examples

The following example shows how to generate the DSA host key pair.

```
device(config)# crypto key generate dsa
```

The following example shows how to generate the RSA key pair.

```
device(config)# crypto key generate rsa modulus 2048
```

# crypto key zeroize

Deletes the crypto host key pair from the flash memory.

## Syntax

**crypto key zeroize** [ **dsa** | **rsa** ]

## Command Default

SSH is not enabled and the host key pair is saved in the flash memory.

## Parameters

**dsa**

Deletes the DSA host key pair.

**rsa**

Deletes the RSA host key pair.

## Modes

Global configuration mode

## Usage Guidelines

When a host key is generated, it is saved to the flash memory of all management modules. The time to initially generate SSH keys varies depending on the configuration, and can be from a under a minute to several minutes. To disable SSH, you delete all of the host keys from the device. When a host key is deleted, it is deleted from the flash memory of all management modules.

## Examples

The following example shows how to delete the DSA key pair.

```
device(config)# crypto key zeroize dsa
```

The following example shows how to delete the RSA key pair.

```
device(config)# crypto key zeroize rsa
```

The following example shows how to delete DSA and RSA key pairs from flash memory.

```
device(config)# crypto key zeroize
```

## History

| Release version | Command history |
|---|---|
| 5.9.00 | This command was modified. The **cr** option was removed. |

# crypto-ssl certificate

Generates or deletes a crypto SSL certificate.

## Syntax

**crypto-ssl certificate** { **generate** | **zeroize** }

## Parameters

**generate**

Generates an SSL certificate.

**zeroize**

Deletes the currently operative SSL certificate.

## Modes

Global configuration mode

## Usage Guidelines

To allow web management access through HTTPS, you must generate the SSL certificate in addition to enabling web management.

## Examples

The following example shows how to generate a crypto SSL certificate.

```
device(config)# crypto-ssl certificate generate
```

The following example shows how to delete a crypto SSL certificate.

```
device(config)# crypto-ssl certificate zeroize
```

# cycle-time

Sets a limit as to how many seconds users have to be authenticated by Web Authentication.

## Syntax

**cycle-time** *seconds*

**no cycle-time** *seconds*

## Command Default

The default is 600 seconds.

## Parameters

*seconds*

Specifies the authentication cycle time. Valid values are from 0 through 3600 seconds. If the value is set to 0, then there is no limit.

## Modes

Web Authentication configuration mode

## Usage Guidelines

You can set a limit as to how many seconds users have to be authenticated by the Web Authentication by defining a cycle time. This time begins upon the first Login attempt by the user on the Login page. If the user has not been authenticated successfully when this time expires, the user must enter a valid URL again to display the Web Authentication Welcome page.

The **no** form of the command resets the time to the default.

## Examples

The following example sets the cycle time to 100 seconds.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# cycle-time 100
```

# Commands D through H

## dampening

Sets dampening parameters for the route in BGP address-family mode.

### Syntax

**dampening** { *half-life reuse suppress max-suppress-time* | **route-map** *route-map* }

**no dampening**

### Parameters

*half-life*

Number of minutes after which the route penalty becomes half its value. Range is from 1 through 45. Default is 15.

*reuse*

Minimum penalty below which the route becomes usable again. Range is from 1 through 20000. Default is 750.

*suppress*

Maximum penalty above which the route is suppressed by the device. Range is from 1 through 20000. Default is 2000.

*max-suppress-time*

Maximum number of minutes a route can be suppressed by the device. Default is 40.

*route-map*

Enables selection of dampening values established in a route map by means of the **route-map** command.

*route-map*

Name of the configured route map.

### Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

### Usage Guidelines

Use the **no** form of this command to disable dampening.

Use **dampening** without operands to set default values for all dampening parameters.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

To use the dampening values established in a route map, configure the route map first, and then enter the **route-map** command, followed by the name of the configured route map.

A full range of dampening values (*half-life*, *reuse*, *suppress*, *max-suppress-time*) can also be set by means of the **set as-path prepend** command.

# Examples

This example enables default dampening as an IPv4 address-family function.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# dampening
```

This example changes all the dampening values as an IPv6 address-family function.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# dampening 20 200 2500 40
```

This example applies the dampening half-life established in a route map, configures the route map using the **set dampening** command.

```
device# configure terminal
device(config)# route-map myroutemap permit 1
device(config-route-map myroutemap)# set dampening 20
```

# database-overflow-interval (OSPFv2)

Configures frequency for monitoring database overflow.

## Syntax

**database-overflow-interval** *interval*

**no database-overflow-interval**

## Command Default

0 seconds. If the device enters OverflowState, you must reboot before the device leaves this state.

## Parameters

*interval*

Time interval at which the device checks to see if the overflow condition has been eliminated. Valid values range from 0 through 86400 seconds.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

This command specifies how long a device that has entered the OverflowState waits before resuming normal operation of external LSAs. However, if the external link state database (LSDB) is still full, the device lapses back into OverflowState. If the configured value of the database overflow interval is zero, then the device never leaves the database overflow condition.

When the maximum size of the LSDB is reached (this is a configurable value in the *external-lsdb-limit* CLI), the device enters OverflowState. In this state, the device flushes all non-default AS-external-LSAs that the device had originated. The devie also stops originating any non-default external LSAs. Non-default external LSAs are still accepted if there is space in the database after flushing. If no space exists, the Non-default external LSAs are dropped and not acknowledged.

The **no** form of the command disables the overflow interval configuration.

## Examples

The following example configures a database-overflow interval of 60 seconds.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# database-overflow-interval 60
```

# database-overflow-interval (OSPFv3)

Configures frequency for monitoring database overflow.

## Syntax

**database-overflow-interval** *interval*

**no database-overflow-interval**

## Command Default

10 seconds. If the router enters OverflowState, you must reboot before the router leaves this state.

## Parameters

*interval*

Time interval at which the device checks to see if the overflow condition has been eliminated. Valid values range from 0 through 86400 seconds (24 hours).

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

This command specifies how long after a router that has entered the OverflowState before it can resume normal operation of external LSAs. However, if the external link state database (LSDB) is still full, the router lapses back into OverflowState.

When the maximum size of the LSDB is reached (this is a configurable value in the *external-lsdb-limit* CLI), the router enters OverflowState. In this state, the router flushes all non-default AS-external-LSAs that the router had originated. The router also stops originating any non-default external LSAs. Non-default external LSAs are still accepted if there is space in the database after flushing. If no space exists, the Non-default external LSAs are dropped and not acknowledged.

If the configured value of the database overflow interval is 0, then the device never leaves the database overflow condition.

The **no** form of the command disables the overflow interval configuration.

## Examples

The following example configures a database-overflow interval of 120 seconds.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# database-overflow-interval 120
```

# dead-interval

Configures the interval for which a Virtual Router Redundancy Protocol (VRRP) backup router waits for a hello message from the VRRP master router before determining that the master is offline. When backup routers determine that the master is offline, the backup router with the highest priority becomes the new VRRP master router.

## Syntax

**dead-interval** [ **msec** ] *interval*

**no dead-interval** [ **msec** ] *interval*

## Command Default

The default dead interval is internally derived from the hello interval. It is equal to 3 times the hello interval plus the skew time, where the skew time is equal to (256 minus the priority) divided by 256.

## Parameters

**msec** *interval*

Sets the interval, in milliseconds, for which a VRRP backup router waits for a hello message from the VRRP master router before determining that the master is offline. Valid values range from 100 through 84000. The default value is 1000. VRRP-E does not support the dead interval in milliseconds.

*interval*

Sets the interval, in seconds, for which a VRRP backup router waits for a hello message from the VRRP master router before determining that the master is offline. Valid values range from 1 through 84. The default value is 1.

## Modes

VRID interface configuration mode

## Usage Guidelines

By default, the dead interval is internally derived from the hello interval. It is equal to 3 times the hello interval plus the skew time, where the skew time is equal to (256 minus the priority) divided by 256. Generally, if you change the hello interval on the VRRP master device using the **hello-interval** command, you should also change the dead interval on the VRRP backup devices using the **dead-interval** command.

A VRRP master router periodically sends hello messages to the backup routers. The backup routers use the hello messages as verification that the master is still online. If the backup routers stop receiving the hello messages for the period of time specified by the dead interval, the backup routers determine that the master router is offline. At that point, the backup router with the highest priority becomes the new master router.

The **dead-interval** command is configured only on VRRP backup routers and is supported by VRRP and VRRP-E.

The **no** form resets the dead interval to its default value of 1000 milliseconds (1 second).

**NOTE**
VRRP-E does not support the hello message interval in milliseconds.

# Examples

The following example sets a waiting period of 25000 milliseconds before a VRRP backup router determines that a VRRP master router is offline.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# backup priority 40 track-priority 10
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.5.99
device(config-if-e1000-1/1/6-vrid-1)# dead-interval msec 25000
device(config-if-e1000-1/1/6-vrid-1)# activate
```

The following example sets a waiting period of 25 seconds before a VRRP-E backup router determines that a VRRP master router is offline.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/5
device(conf-if-e1000-1/1/5)# ip address 10.53.5.3/24
device(conf-if-e1000-1/1/5)# ip vrrp-extended vrid 2
device(conf-if-e1000-1/1/5-vrid-2)# backup priority 50 track-priority 10
device(conf-if-e1000-1/1/5-vrid-2)# ip-address 10.53.5.1
device(conf-if-e1000-1/1/5-vrid-2)# dead-interval 25
device(conf-if-e1000-1/1/5-vrid-2)# activate
```

# dead-interval (VSRP)

Configures the number of seconds a backup waits for a Hello message from the master before determining that the master is dead.

## Syntax

**dead-interval** *number*

**no dead-interval** *number*

## Command Default

The default time interval for the backup to wait for the Hello message from the master is 3 seconds.

## Parameters

*number*

Specifies the time interval for which the backup waits for the Hello message from the master. The time interval range is from 1 through 84 seconds.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The **no** form of the command resets the time interval to the default value.

## Examples

The following example shows how to change the dead interval.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# dead-interval 30
```

# decnet-proto

Configures the DECnet protocol VLAN.

## Syntax

**decnet-proto** [ **name** *string* ]

**no decnet-proto** [ **name** *string* ]

## Command Default

The DECnet protocol VLAN is not configured.

## Parameters

**name** *string*

Specifies the name of the DECnet protocol VLAN that you want to configure. The name can be up to 32 characters in length.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the DECnet protocol VLAN.

## Examples

The following example shows how to configure a DECnet protocol VLAN.

```
device(config)# vlan 2
device(config-vlan-2)# decnet-proto name Red
device(config-vlan-decnet-proto)# no dynamic
```

# default-acl

Configures the default ACL for failed, timed-out, or guest user sessions.

## Syntax

**default-acl** { **ipv4** | **ipv6** } [ *acl-id* |*acl-name* ] [ **in** | **out** ]

**no default-acl** { **ipv4** | **ipv6** } [ *acl-id* |*acl-name* ] [ **in** | **out** ]

## Parameters

**ipv4**
Specifies an IPv4 AC.

**ipv6**
Specifies an IPv6 ACL.

*acl-id*
ID of standard or numbered ACL (IPv4 only).

*acl-name*
Name or extended name of the ACL.

**in**
Specifies incoming authentication.

**out**
Specifies outgoing authentication.

## Modes

Flexible-authentication configuration sub-mode

## Usage Guidelines

Use the **no** form of the command to remove the configurable default ACL.

Use the command to configure a default ACL to be applied to users who failed (restricted VLAN), timed out (critical VLAN), or are guests (not capable of dot1x authentication).

**NOTE**
Dynamic modification of a default ACL by adding or deleting ACL rules is not supported. To modify a default ACL, you must first clear the session.

## Examples

The following example configures the default IPv4 ACL called guests for inbound authentication.

```
device# configure terminal
device(conf)# authentication
device(conf-authen)# default-acl ipv4 guest in
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# default-gateway

Configures the default gateway for a VLAN.

## Syntax

**default-gateway** *ip-address metric*

**no default-gateway** *ip-address metric*

## Command Default

The default gateway is not configured.

## Parameters

*ip-address*

Specifies the IP address of the gateway router.

*metric*

Specifies the metric (cost) of the gateway. You can specify a value from 1 through 5. There is no default. The gateway with the lowest metric is used.

## Modes

VLAN configuration mode

## Usage Guidelines

You can configure up to five default gateways for the designated VLAN, and associate a metric with each one. The software uses the gateway with the lowest metric. The other gateways reside in the configuration but are not used. To use one of the other gateways, modify the configuration so that the gateway you want to use has the lowest metric. If more than one gateway has the lowest metric, the gateway that appears first in the running-config is used.

If you have already configured a default gateway globally using the **ip default-gateway** command and you do not configure a gateway in the VLAN, the software uses the globally configured gateway and gives the gateway a metric value of 1.

The **no** form of the command removes the gateway configuration for a VLAN.

## Examples

The following example shows how to set the default gateway for a management VLAN. Because the 10.10.10.1 gateway has a lower metric, the software uses this gateway. The other gateway remains in the configuration, but is not used. You can use the other one by changing the metrics so that the 10.20.20.1 gateway has the lower metric.

```
device(config)# vlan 10
device(config-vlan-10)# default-gateway 10.10.10.1 1
device(config-vlan-10)# default-gateway 10.20.20.1 2
```

# default-information-originate (BGP)

Configures the device to originate and advertise a default BGP4 or BGP4+ route.

## Syntax

**default-information-originate**

**no default-information-originate**

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

This example originates and advertises a default BGP4 route for the default VRF.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# default-information-originate
```

This example originates and advertises a default BGP4 route for VRF “red”

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# default-information-originate
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# default-information-originate (OSPFv2)

Controls distribution of default information to an OSPFv2 device.

## Syntax

**default-information-originate** [ **always** ] [ **metric** *metric* ] [ **metric-type** { **type1** | **type2** } ] [ **route-map** *name* ]

**no default-information-originate**

## Command Default

The default route is not advertised into the OSPFv2 domain.

## Parameters

**always**

Always advertises the default route. If the route table manager does not have a default route, the router advertises the route as pointing to itself.

**metric** *metric*

Specifies the cost for reaching the rest of the world through this route. If you omit this parameter and do not specify a value using the *default-metric* router configuration command, a default metric value of 10 is used. Valid values range from 1 through 65535. The default is 10.

**metric-type**

Specifies how the cost of a neighbor metric is determined. The default is **type1**. However, this default can be changed with the **metric-type** command.

**type1**

Type 1 external route.

**type2**

Type 1 external route,

**route-map** *name*

Specifies that the default route is generated if the route map is satisfied. This parameter overrides other options. If the **set metric** and **set metric-type** commands are specified in the route-map, the command-line values of metric and metric-type if specified, are “ignored” for clarification.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

This configuration provides criteria for the redistribution of any default routes found in the route table manager (RTM),whether static or learned from another protocol, to its neighbors.

The corresponding route-map should be created before configuring the **route-map** option, along with the **default-information-originate** command. If the corresponding route-map is not created beforehand, an error message is displayed stating that the route-map must be created.

The route-map option cannot be used with a non-default address in the match conditions. The default route LSA is not generated if a default route is not present in the routing table and a **match ip address** condition for an existing non-default route is configured in the route-map. The **match ip address** command in the route-map is a no-op operation for the default information originate command.

A device does not inject the default route into an NSSA by default and this command does not cause the device to inject the default route into the NSSA. To inject the default route into an NSSA, use the **area nssa default-information-originate** command.

The **no** form of the command disables default route origination.

# Examples

The following example creates and advertises a default route with a metric of 30 and a type 1 external route.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# default-information-originate metric 30 metric-type type1
```

# default-information-originate (OSPFv3)

Controls distribution of default information to an OSPFv3 device.

## Syntax

**default-information-originate** [ **always** ] [ **metric** *metric* ] [ **metric-type** { **type1** | **type2** } ]

**no default-information-originate**

## Command Default

The default route is not advertised into the OSPFv3 domain.

## Parameters

**always**

Always advertises the default route. If the route table manager (RTM) does not have a default route, the router advertises the route as pointing to itself.

**metric** *metric*

Used for generating the default route, this parameter specifies the cost for reaching the rest of the world through this route. If you omit this parameter, the value of the **default-metric** command is used for the route. Valid values range from 1 through 65535.

**metric-type**

Specifies the external link type associated with the default route advertised into the OSPF routing domain.

**type1**

The metric of a neighbor is the cost between itself and the router plus the cost of using this router for routing to the rest of the world.
If you do not use this option, the default redistribution metric type is used for the route type.

**type2**

The metric of a neighbor is the total cost from the redistributing routing to the rest of the world.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

This configuration provides criteria for the redistribution of any default routes found in the RTM (whether static or learned from another protocol) to its neighbors.

If you specify a metric and metric type, the values are used even if you do not use the always option.

The **no** form of the command disables default route origination.

# Examples

The following example specifies a metric of 20 for the default route redistributed into the OSPFv3 routing domain and an external metric type of Type 2.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# default-information-originate metric 20 metric-type type2
```

# default-ipv6-gateway

Configures the IPv6 address of the default gateway on a VLAN.

## Syntax

**default-ipv6-gateway** *ipv6-address* [*metric* ]

**no default-ipv6-gateway**

## Parameters

*ipv6-address*

IPv6 address of the default gateway.

*metric*

A decimal value from 1 through 5.

## Modes

VLAN configuration mode

## Usage Guidelines

A device should have a default gateway, for the following reasons:

- Although IPv6 discovers neighbors and routes dynamically, in some cases Router Advertisement (RA) and Router Solicitation (RS) operations are disabled and a default gateway is required to send traffic. RA and RS are not suppressed if a default gateway is configured.
- Management devices (for example, TFTP servers, Telnet or SSH clients) are not members of the same subnet as the management IPv6 address.

If a management VLAN is not configured, the device can have only one IPv6 default gateway in the global configuration.

If a management VLAN is configured, the device can have a maximum of 5 IPv6 default gateways, with an optional metric (1 through 5), under the management VLAN. Multiple gateways can have the same metric value.

Configured gateway addresses and the default gateway address must be in same subnet.

The best default gateway is first chosen as the device whose neighbors are reachable (in the REACH state), in the sequence of metric values. Otherwise, the gateway with the highest priority (the lowest metric value) is chosen.

If a static default gateway is configured, that gateway takes precedence over the best default gateway configured by means of RA. If the static default-gateway configuration is removed, the best default gateway learned by RA is restored.

Use the **no** form of the command to remove the IPv6 address and disable the default gateway.

Selection of the best default router among configured IPv6 routers occurs under the following conditions:

- Disabling an interface
- Processing of an NA message receipt
- Adding or deleting an IPv6 neighbor to or from the neighbor list
- Configuring the IPv6 static default gateway by means of the CLI

The process of resolving the link layer for the IPv6 default gateway by sending NS occurs during the following conditions:

- Configuration of the default gateway configured by means of the CLI
- Addition or deletion of a management VLAN configuration

# Examples

The following example configures the maximum of 5 IPv6 default gateways with the management VLAN configuration, and specifies metrics for each.

```
device# configure terminal
device(config)# vlan 66
device(config-vlan-66)# default-ipv6-gateway 2620:100:c:fe23:10:37:65:129 3
device(config-vlan-66)# default-ipv6-gateway 2620:100:c:fe23:10:37:65:129 2
device(config-vlan-66)# default-ipv6-gateway 2620:100:c:fe23:10:37:65:130 2
device(config-vlan-66)# default-ipv6-gateway 2620:100:c:fe23:10:37:65:131 1
device(config-vlan-66)# default-ipv6-gateway 2620:100:c:fe23:10:37:65:132 5
```

Use the **show ipv6** command to confirm the configuration and view the best default gateway (router).

```
device(config)# show ipv6
Global Settings
  IPv6 is enabled
  Link-local address(es):
    fe80::768e:f8ff:fef9:6d80 [Preferred]
  Global unicast address(es):
    2620:100:c:fe23:768e:f8ff:fef9:6d80 [Preferred],  subnet is 2620:100:c:fe23::/64
  Joined group address(es):
    ff02::1:fff9:6d80
    ff02::1
  Best Default Router : 2620:100:c:fe23:10:37:65:129 PMTUS : 0
  MTU is 1500 bytes
  ND DAD is enabled, number of DAD attempts: 3
  ND reachable time is 30000 miliseconds
  ND retransmit interval is 1000 miliseconds
  Current Hop Limit is 64
  Hosts use stateless autoconfig for addresses
  No Inbound Access List Set
  No Outbound Access List Set
  No IPv6 Domain Name Set
  No IPv6 DNS Server Address set
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# default-local-preference

Enables setting of a local preference value to indicate a degree of preference for a route relative to that of other routes.

## Syntax

**default-local-preference** *num*

**no default-local-preference**

## Parameters

*num*

Local preference value. Range is from 0 through 65535. The default is 100.

## Modes

BGP configuration mode

## Usage Guidelines

Local preference indicates a degree of preference for a route relative to that of other routes. BGP4 neighbors can send the local preference value as an attribute of a route in an UPDATE message.

## Examples

The following example sets the local preference value to 200.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# default-local-preference 200
```

# default-metric (BGP)

Changes the default metric used for redistribution.

## Syntax

**default-metric** *value*

**no default-metric**

## Command Default

The default metric value is 1.

## Parameters

*value*

Metric value. Range is from 0 through 65535. The default metric value is 1.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

Use the **no** form of this command to restore the default.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

This example changes the default metric used for redistribution to 100.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# default-metric 100
```

# default-metric (OSPF)

Sets the default metric value for the OSPFv2 or OSPFv3 routing protocol.

## Syntax

**default-metric** *metric*

**no default-metric**

## Parameters

*metric*

OSPF routing protocol metric value. Valid values range from 1 through 65535. The default is 10.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

This command overwrites any incompatible metrics that may exist when OSPFv2 or OSPFv3 redistributes routes. Therefore, setting the default metric ensures that neighbors will use correct cost and router computation.

The **no** form of the command restores the default setting.

## Examples

The following example sets the default metric to 20 for OSPF.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# default-metric 20
```

# default-metric (RIP)

Changes the RIP metric the router assigns by default to redistributed routes.

## Syntax

**default-metric** *value*

**no default-metric** *value*

## Command Default

By default, a metric of 1 is assigned to each route that is redistributed into RIP.

## Parameters

*value*

Specifies a numeric value from 1 through 15 that is assigned to each route redistributed into RIP.

## Modes

RIP router configuration mode.

## Usage Guidelines

The **no** form of the command returns the value of the default-metric to 1.

As its default-metric increases, the less likely a route is to be used.

## Examples

The following example sets the default metric for all RIP routes on the device to 10.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# default-metric 10
```

The following example returns the default metric set in the previous example to the system default (1).

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# no redistribute connected metric 10
```

# default-passive-interface

Marks all OSPFv2 and OSPFv3 interfaces passive by default.

## Syntax

**default-passive-interface**

**no default-passive-interface**

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

When you configure the interfaces as passive, the interfaces drop all the OSPFv2 and OSPFv3 control packets.

You can use the **ip ospf active** and **ip ospf passive** commands in interface subconfiguration mode to change active/passive state on specific OSPFv2 interfaces. You can use the **ipv6 ospf active** and **ipv6 ospf passive** commands in interface subconfiguration mode to change the active and passive state on specific OSPFv3 interfaces.

The **no** form of the command disables the passive state.

## Examples

The following example marks all OSPFv2 interfaces as passive.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# default-passive-interface
```

The following example marks all OSPFv3 interfaces as passive for VRF “red”.

```
device# configure terminal
device(config)# ipv6 router ospf vrf red
device(config-ospf6-router-vrf-red)# default-passive-interface
```

# default-ports

Assigns ports (interfaces) other than the factory-assigned ports as the default stacking ports.

## Syntax

**default-ports** *unit**/**slot**/** port*

**no default-ports**

## Command Default

The factory-assigned default stacking ports are the only default stacking ports on the device.

## Parameters

*unit*
: Stack unit ID for the device on which the interface resides.

*slot*
: Stack unit slot or module on which the interface resides.

*port*
: Interface to be used as a default stacking port.

## Modes

Stack unit configuration mode

## Usage Guidelines

The **no** form of the command restores the factory-assigned default stacking ports. Any ports you previously assigned as the default stacking ports using the **default-ports** command are overwritten.

When you use the **default-ports** command, the factory-assigned default stacking ports are no longer the default stacking ports.

Only valid stacking ports can be assigned as default stacking ports. Valid ports vary depending on the type of FastIron device.

Tagged ports cannot be assigned as default stacking ports.

The number of ports you can assign as default stacking ports varies depending on the type of FastIron device. Some devices allow you to assign two ports as the default stacking ports, and some devices allow you to assign a single port as the default stacking port.

Default ports cannot be changed on ICX 7150 devices.

# Examples

The following example assigns the stacking ports on Module 3 on the rear panel of an ICX 7750 as the default stacking ports.

```
device# configure terminal
device(config)# stack unit 1
device¡(config-unit-1)# default-ports 1/3/1 1/3/4
```

# default-timers

Resets the GVRP Join, Leave, and Leaveall timers to the default values.

## Syntax

**default-timers**

## Command Default

The default value for the Join timer is 200 ms. The default value for the Leave timer is 600 ms. The default value for the Leaveall timer is 10,000 ms.

## Modes

GVRP configuration mode

## Usage Guidelines

You can use the **join-timer** command to change the values of these timers.

## Examples

The following example shows how to reset the timers to the default values.

```
device(config)# gvrp-enable
device(config-gvrp)# default-timers
```

# default-vlan-id

Changes the default VLAN ID.

## Syntax

**default-vlan-id** *vlan-id*

**no default-vlan-id** *vlan-id*

## Command Default

The default VLAN ID is 1.

## Parameters

*vlan-id*

Specifies the VLAN ID that you want to configure as the default. Valid VLAN ID values are from 1 through 4095.

## Modes

Global configuration mode

## Usage Guidelines

You must specify a valid VLAN ID that is not already in use. For example, if you have already defined VLAN 10, you cannot use "10" as the new VLAN ID for the default VLAN.

**NOTE**
This command does not change the properties of the default VLAN. Changing the name allows you to use the VLAN ID "1" as a configurable VLAN.

The **no** form of the command resets the VLAN ID to the default.

## Examples

The following example shows how to change the default VLAN ID.

```
device(config)# default-vlan-id 4095
```

# delay-notifications

Configures the delay time for notifying the Layer 3 protocols of the VE down event.

## Syntax

**delay-notifications** *value*

**no delay-notifications** *value*

## Command Default

The delay time is not configured.

## Parameters

*value*

The time to delay the notification of the VE down event. The value can range from 1 through 60 seconds.

## Modes

VE interface configuration mode

## Usage Guidelines

When all the ports in the VLAN go into the non-forwarding state, the device waits for the configured time before notifying the Layer 3 protocols of the VE down event. Once the timer expires, if the ports remain in the non-forwarding state, the device notifies the Layer 3 protocols of the VE down event.

If any of the ports comes into the forwarding state before the timer expires, the device cancels the existing timer for the VE down event.

The **no** form of the command removes the configured delay time.

## Examples

The following example shows configuring the delay time on interface 50 to 20 seconds.

```
device(config)#  interface ve 50
device(config-vif-50)# delay-notifications 20
```

## History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was introduced. |

# delete-all

Deletes all user records from a local user database.

## Syntax

**delete-all**

## Modes

Local user database configuration mode

## Examples

The following example deletes all user records from the local user database "localdb1".

```
device(config)# local-userdb localdb1
device(config-localuserdb-localdb1)# delete-all
```

# deny (extended IPv4 ACLs)

Inserts filtering rules in IPv4 extended named or numbered ACLs that will deny packets.

## Syntax

Use the following syntax to define a TCP or UDP rule that will deny packets:

[ **no** ] **deny** { **tcp** | **udp** } { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ *source-comparison-operators* ] { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ **established** ] [ *destination-comparison-operators* ] [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define an ICMP rule that will deny packets:

[ **no** ] **deny icmp** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ *icmp-num* | *icmp-type* ] [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define a rule for protocols other than TCP, UDP, or ICMP that will deny packets:

[ **no** ] **deny** *ip-protocol* { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

**no sequence** *seq-num*

## Parameters

*ip-protocol*

Specifies the type of IPv4 packet to filter. You can either specify a protocol number (from 0 through 255) or a supported protocol name. For a complete list of protocols, type **?** after **deny**. Supported protocols include:

- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **igrp**—Internet Gateway Routing Protocol
- **ip**—any IPv4 protocol
- **ospf**—Open Shortest Path First
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

*S_IPaddress*

Specifies a source address for which you want to filter the subnet.

*mask*

Defines a mask, whose effect is to specify a subnet that includes the source address that you specified. For options to specify the mask, see the Usage Guidelines.

**host**

Specifies the source as a host.

*S_IPaddress*

Specifies the source address of the host.

**any**

Specifies all source addresses.

*source-comparison-operators* and *destination-comparison-operators*

If you specified **tcp** or **udp**, the following optional operators are available:

**eq**

Specifies the address is equal to the port name or number you enter after **eq**.

**gt**

Specifies port numbers that are equal to or greater than the port number or that are equal to or greater than the numeric equivalent of the port name you enter after **gt**.

**lt**

Specifies port numbers that are equal to or less than the port number or that are equal to or less than the numeric equivalent of the port name you enter after **lt**.

**neq**

Specifies all port numbers except the port number or port name you enter after **neq**.

**range**

Specifies all port numbers that are between the first port name or number and the second name or number you enter following the **range** keyword. Enter the range as two values separated by a space. The first port number in the range must be less than the last number in the range. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: 23 53 .

*D_IPaddress*

Specifies a destination address for which you want to filter the subnet.

*mask*

Defines a subnet mask that includes the destination address that you specified. For mask options, refer to the Usage Guidelines.

**host**

Specifies a host as destination.

*D_IPaddress*

Specifies the destination address of the host.

**any**

Specifies all destination addresses.

**established**

(For TCP rules only) Filter packets that have the Acknowledgment (ACK) or Reset (RST) flag set. This policy applies only to established TCP sessions, not to new sessions.

*icmp-num* | *icmp-type*

(For ICMP only) Specifies a named or numbered message type.

*icmp-num*

Specifies a numbered message type. Use this format if the rule also needs to include **precedence**, **tos** , one of the DSCP options, one of the 802.1p options, **internal-priority-marking** , or **traffic-policy**.

**any-icmp-type**
Specifies any ICMP type.

**echo**
Specifies an echo request (ping).

**echo-reply**
Specifies an echo reply.

**information-request**
Specifies an information request.

**mask-reply**
Specifies an address mask reply.

**mask-request**
Specifies an address mask request.

**parameter-problem**
Specifies a parameter problem.

**redirect**
Specifies a redirect message.

**source-quench**
Specifies a relieve congestion message.

**time-exceeded**
Specifies a time exceeded message.

**timestamp-reply**
Specifies a timestamp reply.

**timestamp-request**
Specifies a timestamp request.

**unreachable**
Specifies a destination-unreachable message.

**precedence** { *precedence-name* | *precedence-value* }
Specifies a *precedence-name* or corresponding *precedence-value*, as follows:

**0** or **routine**
Specifies routine precedence.

**1** or **priority**
Specifies priority precedence.

**2** or **immediate**
Specifies immediate precedence.

**3** or **flash**
Specifies flash precedence.

**4** or **flash-override**
Specifies flash-override precedence.

**5** or **critical**
Specifies critical precedence.

**6** or **internet**
Specifies internetwork control precedence.

**7** or **network**
Specifies network control precedence.

**tos** { *tos-name* | *tos-value* }
Specifies a type of service (ToS). Enter either a supported *tos-name* or the equivalent *tos-value*.

**0** or **normal**
Specifies normal ToS.

**1** or **min-monetary-cost**
Specifies min monetary cost ToS.

**2** or **max-reliability**
Specifies max reliability ToS.

**4** or **max-throughput**
Specifies max throughput ToS.

**8** or **min-delay**
Specifies min-delay ToS.

**dscp-matching** *dscp-value*
Filters by DSCP value. Values range from 0 through 63.

**dscp-marking** *dscp-value*
Assigns the DSCP value that you specify to the packet. Values range from 0 through 63.

**802.1p-priority-matching** *802.1p-value*
Filters by 802.1p priority, for rate limiting. Values range from 0 through 7.

**802.1p-priority-marking** *802.1p-value*
Assigns the 802.1p value that you specify to the packet. Values range from 0 through 7.

**internal-priority-marking** *queuing-priority*
Assigns the internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**802.1p-and-internal-marking** *priority-value*
Assigns the identical 802.1p value and internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**traffic-policy** *name*
Enables the device to limit the rate of inbound traffic and to count the packets and bytes per packet to which ACL deny clauses are applied. For configuration procedures and examples, refer to the chapter "Traffic Policies" in the *Ruckus FastIron Traffic Management Configuration Guide*.

**log**
Enables SNMP traps and Syslog messages for the rule. In addition, logging must be enabled using the **acl-logging** command.

**mirror**
Mirrors packets matching the rule.

# Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

# Usage Guidelines

Extended ACLs deny traffic according to source and destination addresses, port protocol, and other IPv4 frame content. You can also enable logging and mirroring.

The order of the rules in an ACL is critical, as the first matching rule stops further processing.

You can specify a mask in either of the following ways:

- Wildcard mask format (for example, 0.0.0.255). The advantage of this format is that it enables you mask any bit, for example by specifying 0.255.0.255.
- Classless Interdomain Routing (CIDR) format, in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 in the wildcard mask format.

If you specify **icmp** and also specify the **any-icmp-type** option, the following QoS options are not available: **dscp-marking**, **dscp-matching**, **internal-priority-marking**, **802.1p-priority-marking**, ond **802.1p-priority-matching**.

On the Ruckus ICX 7150 and Ruckus ICX 7750, ACL logging is not supported for egress ACLs.

When specifying type of service (ToS), you can indicate multiple *tos-value* options by entering the sum of the needed ToS options. For example, to specify both **max-reliability** and **min-delay**, enter **10**. To specify all options, enter **15**. Values range from **0** through **15**.

In a rule that includes one or more of the following parameters, the **log** keyword is ignored:

- **dscp-matching**
- **dscp-marking**
- **802.1p-priority-matching**
- **802.1p-priority-marking**
- **802.1p-and-internal-marking**

For details on 802.1p priority matching, refer to "Inspecting the 802.1p bit in the ACL for adaptive rate limiting" in the *Ruckus FastIron Traffic Management Configuration Guide*.

To delete a deny rule from an ACL, type **no** followed by the full command syntax.

# Examples

The following ACL, applied to an Ethernet interface, blocks and logs IPv4 TCP packets transmitted by Telnet from a specified host.

```
device# configure terminal
device(config)# ip access-list extended "block Telnet"
device(config-ext-nacl)# deny tcp host 10.157.22.26 any eq telnet log
device(config-ext-nacl)# exit
device(config)# interface ethernet 1/1/1
device(config-if-1/1/1)# ip access-group "block Telnet" in
```

# deny (standard IPv4 ACLs)

Inserts filtering rules in IPv4 standard named or numbered ACLs that will deny packets.

## Syntax

**deny** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

**no deny** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

## Parameters

*S_IPaddress*
: Specifies a source address for which you want to filter the subnet.

*mask*
: Defines a subnet mask that includes the source address you specified.

**host**
: Indicates the source IP address is a host address.

*S_IPaddress*
: Specifies source address.

**any**
: Specifies all source addresses.

**log**
: Enables logging for the rule.

**mirror**
: Mirrors packets matching the rule.

## Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

## Usage Guidelines

This command configures rules to drop traffic based on source addresses. You can also enable logging and mirroring.

Standard ACLs deny traffic according to source address only.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list. Such a rule is automatically assigned the next multiple of 10 as a sequence number.

You can specify a mask in either of the following ways:

- Wildcard mask format. The advantage of this format is that it enables you to mask any bit, for example by specifying 0.255.0.255.

- Classless Interdomain Routing (CIDR) format—in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 in the wildcard mask format.

On Ruckus ICX 7150 and Ruckus ICX 7750 devices, ACL logging is not supported for egress ACLs.

For the **log** keyword to trigger a log entry, logging must be enabled with the **acl-logging** command.

To delete a rule from an ACL, use the **no deny** command followed by the full command syntax.

# Examples

The following example shows how to configure a standard numbered ACL and apply it to incoming traffic on port 1/1/1.

```
device# configure terminal
device(config)# ip access-list standard 1
device(config-std-nacl)# deny host 10.157.22.26 log
device(config-std-nacl)# deny 10.157.29.12 log
device(config-std-nacl)# deny host IPHost1 log
device(config-std-nacl)# permit any
device(config-std-nacl)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip access-group 1 in
```

# description (IKEv2)

Describes an Internet Key Exchange version 2 (IKEv2) profile.

## Syntax

**description** *text-string*

**no description**

## Command Default

An IKEv2 profile description is not configured.

## Parameters

*text-string*

Specifies the IKEv2 profile description. The string must be from 1 through 64 ASCII characters in length.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

Configuring a profile description is optional, but in a complex network configuration with a number of IKEv2 profiles, a profile description can help to identify a specific profile.

The **no** form of the command removes the IKEv2 profile description.

## Examples

The following example shows how to configure a description for an IKEv2 profile named prof-dept1.

```
device(config)# ikev2 profile prof_dept1
device(config-ike-profile-prof_dept1)# descripton PersonnelDepartmentUSA
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# description (IPsec)

Describes an IP security (IPsec) profile.

## Syntax

**description** *text-string*

**no description**

## Command Default

An IPsec profile description is not configured.

## Parameters

*text-string*

Specifies the IPsec profile description. The string must be from 1 through 64 ASCII characters in length.

## Modes

IPsec profile configuration mode

## Usage Guidelines

Configuring a profile description is optional, but in a complex network configuration with a number of IPsec profiles, a profile description can help to identify a specific profile.

The **no** form of the command removes the IPsec profile description.

## Examples

The following example shows how to configure a description for an IPsec profile named prof-dept2.

```
device(config)# ipsec profile prof-dept2
device(config-ipsec-profile-prof-dept2)# description FinanceDepartmentCanada
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# destination-ip

Sets the destination IP address of an ERSPAN mirror.

## Syntax

**destination-ip** *ip-addr*

**no destination-ip** *ip-addr*

## Command Default

A destination IP is not configured for the ERSPAN profile.

## Parameters

*ip-addr*

Specifies the IP address in the format A.B.C.D.

## Modes

Monitor profile mode

## Usage Guidelines

The destination IP address is the IP address for the remote host that is collecting the mirrored traffic, not the switch.

The **no** form of the command removes the IP address from the monitor profile.

## Examples

The following example sets the destination IP address in ERSPAN profile 3.

```
device(config)# monitor-profile 3 type ERSPAN
device(config-monitor-profile 3)# destination-ip 1.1.1.1
device(config-monitor-profile 3)# exit
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# dhcp-default-router

Specifies the IP addresses of the default routers for a client.

## Syntax

**dhcp-default-router** *address*

## Parameters

*address*

Specifies the IP address of the default router.

## Modes

DHCP server pool configuration mode

## Examples

The following example specifies the IP address of the default router for a client.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# dhcp-default-router 10.2.1.143
```

# dhcp-gateway-list

Configures a gateway list when DHCP Assist is enabled on a Layer 2 switch.

## Syntax

**dhcp-gateway-list** *num ip-address*

## Parameters

*num*

Specifies the number of the gateway list.

*ip-address*

Specifies the gateway IP address.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

Up to eight addresses can be defined for each gateway list in support of ports that are multi-homed. When multiple IP addresses are configured for a gateway list, the Layer 2 switch inserts the addresses into the discovery packet in a round-robin fashion. Up to 32 gateway lists can be defined for each Layer 2 switch.

## Examples

The following commands configure a gateway list.

```
device(config)# dhcp-gateway-list 1 10.95.5.1
device(config)# dhcp-gateway-list 2 10.95.6.1
device(config)# dhcp-gateway-list 3 10.95.1.1 10.95.5.1
device(config)# interface ethernet 2
device(config-if-e1000-2)# dhcp-gateway-list 1
device(config-if-e1000-2)# interface ethernet 8
device(config-if-e1000-8)# dhcp-gateway-list 3
device(config-if-e1000-8)# interface ethernet 14
device(config-if-e1000-14)# dhcp-gateway-list 2
```

# dhcp snooping client-learning disable

Disables DHCP client learning on an individual port or range of ports.

## Syntax

**dhcp snooping client-learning disable**

**no dhcp snooping client-learning disable**

## Modes

Interface configuration mode.

## Usage Guidelines

Use the **no** form of the command to re-enable DHCP client learning on a port once it has been disabled.

## Examples

The following example disables DHCP client learning on an individual port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# dhcp snooping client-learning disable
```

The following example disables DHCP client learning on a range of ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/5
device(config-mif-1/1/1-1/1/5)# dhcp snooping client-learning disable
```

## History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was modified to include enabling DHCP client learning on a range of ports. |

# dhcp snooping relay information

Enables DHCP snooping relay information (DHCP Option 82) on an interface.

## Syntax

**dhcp snooping relay information**

**no dhcp snooping relay information**

## Command Default

DHCP option 82 is automatically enabled when DHCP snooping is enabled on the VLAN.

## Modes

Interface configuration mode

## Usage Guidelines

When DHCP snooping is enabled using the **ip dhcp snooping vlan** command, DHCP option 82 is automatically enabled on a VLAN. The **dhcp snooping relay information** command disables or re-enables DHCP option 82 for a specified interface on the VLAN.

The **no** form of the command disables DHCP option 82 for an interface.

## Examples

The following example disables DHCP option 82 on a specified interface after it was automatically enabled when DHCP snooping was enabled on the VLAN.

```
device# configure terminal
device(config)# ip dhcp snooping vlan 100
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# no dhcp snooping relay information
```

The following example re-enables DHCP option 82 on a specified interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# dhcp snooping relay information
```

# dhcp snooping relay information circuit-id

Configures a unique circuit ID per port.

## Syntax

**dhcp snooping relay information circuit-id** *ASCII-string*

**no dhcp snooping relay information circuit-id** *ASCII-string*

## Parameters

*ASCII-string*

Specifies the ASCII-string. The string can be up to 63 characters in length.

## Modes

Interface configuration mode

## Usage Guidelines

**noshow interfaces ethernet**

## Examples

The following example enables the circuit ID per port.

```
device(config)# ip dhcp snooping vlan 1
device(config)# interface ethernet 1/1/4
device(config-if-e1000-1/1/4)# dhcp snooping relay information circuit-id Brcd01
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# dhcp snooping relay information remote-id

Configures a unique remote ID per port.

## Syntax

**dhcp snooping relay information remote-id** *ASCII-string*

**no dhcp snooping relay information remote-id** *ASCII-string*

## Parameters

*ASCII-string*

Specifies the ASCII-string. The string can be up to 63 characters in length.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables the remote ID processing once it is enabled.

Use the **show interfaces ethernet** command to view the remote ID configured on a port.

## Examples

The following example enables the remote ID per port.

```
device(config)# ip dhcp snooping vlan 1
device(config)# interface ethernet 1/1/4
device(config-if-e1000-1/1/4)# dhcp snooping relay information remote-id remote01
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# dhcp snooping relay information subscriber-id

Configures a unique subscriber ID per port or on a range of ports.

## Syntax

**dhcp snooping relay information subscriber-id** *ASCII-string*

**no dhcp snooping relay information subscriber-id** *ASCII-string*

## Parameters

*ASCII-string*

Specifies the ASCII-string. The string can be up to 50 alphanumeric characters in length.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables SID processing once it is enabled.

Use the **show interfaces ethernet** command to view the subscriber ID configured on a port or a range of ports.

## Examples

The following example enables a unique subscriber ID per port.

```
device(config)# ip dhcp snooping vlan 1
device(config)# interface ethernet 1/1/4
device(config-if-e1000-1/1/4)# dhcp snooping relay information subscriber-id Brcd01
```

The following example enables a unique subscriber ID on a range of ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/5
device(config-mif-1/1/1-1/1/5)# dhcp snooping relay information subscriber-id Brcd01
```

## History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was modified to include enabling a unique subscriber ID on a range of ports. |

# dhcp snooping trust

Enables trust on a port connected to a DHCP server.

## Syntax

**dhcp snooping trust**

**no dhcp snooping trust**

## Command Default

The default trust setting for a port is untrusted.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables the trust setting.

## Examples

The following example sets the trust setting of port 1/1/1 to trusted.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# dhcp snooping trust
```

# dhcp6 snooping trust

Enables trust on a port connected to a DHCPv6 server.

## Syntax

**dhcp6 snooping trust**

**no dhcp6 snooping trust**

## Command Default

The default trust setting for a port is untrusted

## Modes

Interface configuration mode.

## Usage Guidelines

The no form of the command disables trust on the port.

## Examples

The following example enables trust on a port connected to a DHCPv6 server.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# dhcp6 snooping trust
```

# dhgroup

Configures a Diffie-Hellman (DH) group for an Internet Key Exchange version 2 (IKEv2) proposal.

## Syntax

**dhgroup** { **14** | **19** | **20** }

**no dhgroup** { **14** | **19** | **20** }

## Command Default

The default DH group is 20.

## Parameters

**14**

Specifies the 2048-bit modular exponential (MODP) DH group.

**19**

Specifies the 256-bit elliptical curve DH (ECDH) group.

**20**

Specifies the 384-bit ECDH group.

## Modes

IKEv2 proposal configuration mode

## Usage Guidelines

Diffie-Hellman negotiations are a part of the IKEv2 negotiations used to establish a secure communications channel.

Multiple DH groups may be configured for an IKEv2 proposal.

When only one DH group is configured for an IKEv2 proposal, removing it restores the default configuration.

The **no** form of the command removes the specified DH group configuration.

## Examples

The following example configures the 2048-bit MODP DH group (14) for an IKEv2 proposal named ikev2_proposal.

```
device(config)# ikev2 proposal ikev2_proposal
device(config-ikev2-proposal-ikev2_proposal)# dhgroup 14
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# diagnostics (MRP)

Enables diagnostics on a metro ring.

## Syntax

**diagnostics**

**no diagnostics**

## Command Default

Diagnostics are disabled by default.

## Modes

Metro ring configuration mode

## Usage Guidelines

This command is valid only on the master node.

When you enable Metro Ring Protocol (MRP) diagnostics, the software tracks Ring Health Packets (RHPs) according to their sequence numbers and calculates how long it takes an RHP to travel one time through the entire ring. The calculated results have a granularity of 1 microsecond. When you display the diagnostics, the output shows the average round-trip time for the RHPs sent since you enabled diagnostics.

The **no** form of the command disables the diagnostics for the ring.

## Examples

The following example enables the diagnostics for metro ring 1.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-1)# diagnostics
```

# disable (LAG)

Disables the individual ports within a LAG.

## Syntax

**disable port-name** *name*

**disable ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ]

## Command Default

LAG ports are not enabled.

## Parameters

**port-name** *name*
Disables a named port within a LAG.

**ethernet** *unit/slot/port*
Disables the Ethernet port within a LAG.

**to** *unit/slot/port*
Disables a range of ports within a LAG.

**lag** *lag-id*
Disables the LAG virtual interface.

## Modes

LAG configuration mode

## Usage Guidelines

To disable a port belonging to a keep-alive LAG, you must configure the **enable** command from the interface configuration mode.

## Examples

The following example shows how to disable a port within a LAG.

```
device(config)# lag blue static id 1
device(config-lag-blue)# ports ethernet 1/1/1 ethernet 1/1/5
device(config-lag-blue)# disable ethernet 1/3/1
```

The following example shows how to disable a port within a keep-alive LAG.

```
device(config)# lag test keep-alive
device(config-lag-test)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# disable
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# disable (NTP)

Disables NTP client and server mode.

## Syntax

**disable** [ **serve** ]

**no disable serve**

## Command Default

NTP is not enabled.

## Parameters

**serve**

Disables serving the time to clients.

## Modes

NTP configuration mode

## Usage Guidelines

To enable client mode, use the **no disable** command. To enable the client and server mode, use the **no disable serve** command. The **no disable serve** command enables both the client and the server. If the client is already enabled and server is disabled, the **no disable server** enables the server.

If the **serve** keyword is specified, NTP does not serve the time to downstream devices. The **serve** keyword disables the NTP server mode functionalities. If the **serve** keyword is not specified, both NTP client mode and NTP server mode are disabled.

**NOTE**
The **disable** command disables NTP client and server mode; it does not remove the NTP configuration.

The **no** form of the command enables NTP client and server mode.

## Examples

The following example disables the NTP server.

```
device(config)# ntp
device(config-ntp)# disable serve
```

# disable (Port)

Disables a port.

## Syntax

**disable**

## Command Default

A port is enabled (active).

## Modes

Interface configuration mode

## Usage Guidelines

A port can be deactivated (disabled) or activated (enabled) using the **enable** command by selecting the appropriate status.

## Examples

The following example disables or inactivate a port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# disable
```

# disable (VSRP)

Disables the VSRP VRID for a port-based VLAN.

## Syntax

**disable**

## Command Default

The VSRP VRID is disabled by default.

## Modes

VSRP VRID configuration mode

## Examples

The following example shows how to disable the VSRP VRID on a VLAN.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup
device(config-vlan-200-vrid-1)# disable
```

# disable authentication md5

Disables the MD5 authentication scheme for Network Time Protocol (NTP).

## Syntax

**disable authentication md5**

**no disable authentication md5**

## Command Default

If JITC is enabled, the MD5 authentication scheme is disabled. In the standard mode, the MD5 authentication scheme is enabled.

## Modes

Global configuration mode

## Usage Guidelines

In the standard mode, both SHA1 and MD5 authentication schemes are supported. If JITC is enabled, The MD5 authentication for Network Time Protocol (NTP) is disabled by default and the **disable authentication md5** command can be seen in the running configuration. In the JITC mode, only the SHA1 option is available. The SHA1 authentication scheme must be enabled manually to define the authentication key for NTP using the **authentication-key key-id** command.

The **no** form of the command enables the MD5 authentication scheme.

## Examples

The following example disables the MD5 authentication scheme.

```
device(config)# disable authentication md5
```

## History

| Release version | Command history |
|---|---|
| 08.0.20a | This command was introduced. |

# disable-aging

Disables aging of MAC sessions at the global level.

## Syntax

**disable-aging** [ **permitted-mac-only** | **denied-mac-only** ]

**no disable-aging** [ **permitted-mac-only** | **denied-mac-only** ]

## Command Default

Aging of MAC sessions is not disabled.

## Parameters

**permitted-mac-only**

Prevents permitted (authenticated and restricted) sessions from being aged out and ages denied sessions.

**denied-mac-only**

Prevents denied sessions from being aged out, but ages out permitted sessions.

## Modes

Authentication mode

## Usage Guidelines

The **no** form of the command does not disable aging.

Use this command to disable the aging of MAC sessions. Use the **disable-aging** command in the authentication mode and the **authentication disable-aging** command at the interface level. The command entered at the interface level overrides the command entered at the authentication level.

## Examples

The example disables aging for permitted MAC addresses.

```
device(config)# authentication
device(config-authen)# disable-aging permitted-mac-only
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# distance (BGP)

Changes the default administrative distances for eBGP, iBGP, and local BGP.

## Syntax

**distance** *external-distance internal-distance local-distance*

**no distance**

## Parameters

*external-distance*
eBGP distance. Range is from 1 through 255.

*internal-distance*
iBGP distance. Range is from 1 through 255.

*local-distance*
Local BGP4 and BGP4+ distance. Range is from 1 through 255.

## Modes

BGP configuration mode

## Usage Guidelines

To select one route over another according to the source of the route information, the device can use the administrative distances assigned to the sources. The administrative distance is a protocol-independent metric that IP devices use to compare routes from different sources. Lower administrative distances are preferred over higher ones.

## Examples

The following example configures the device to change the administrative distance.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# distance 100 150 200
```

# distance (OSPF)

Configures an administrative distance value for OSPFv2 and OSPFv3 routes.

## Syntax

**distance** { **external** | **inter-area** | **intra-area** } *distance*

**no distance**

## Command Default

The administrative distance value for OSPFv2 and OSPFv3 routes is 110.

## Parameters

**external**
Sets the distance for routes learned by redistribution from other routing domains.

**inter-area**
Sets the distance for all routes from one area to another area.

**intra-area**
Sets the distance for all routes within an area.

*distance*
Administrative distance value assigned to OSPF routes. Valid values range from 1 through 255. The default is 110.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

You can configure a unique administrative distance for each type of OSPF route.

The distances you specify influence the choice of routes when the device has multiple routes from different protocols for the same network. The device prefers the route with the lower administrative distance. However, an OSPFv2 or OSPFv3 intra-area route is always preferred over an OSPFv2 or OSPFv3 inter-area route, even if the intra-area route's distance is greater than the inter-area route's distance.

The **no** form of the commands reverts to the default setting.

# Examples

The following example sets the distance value for all external routes to 125.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# distance external 125
```

The following example sets the distance value for intra-area routes to 80.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# distance intra-area 80
```

The following example sets the distance value for inter-area routes to 90.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# distance inter-area 90
```

# distance (RIP)

Increases the administrative distance that the RIP router adds to routes.

## Syntax

**distance** *num*

**no distance** *num*

## Command Default

The default RIP administrative distance is 120.

## Parameters

*num*

A decimal value from 1 through 255 that designates the administrative distance for all RIP routes.

## Modes

RIP router configuration mode.

## Usage Guidelines

The **no** form of the command returns the administrative distance to the default value of 120.

Routes with lower administrative distance are more likely to be used when administrative distance is used for route comparison.

## Examples

The following example sets the administrative distance for RIP routes to 140.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# distance 140
```

The following example returns the administrative distance for RIP routes set in the previous example to the default of 120.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# no distance 140
```

# distribute-list prefix-list (OSPFv3)

Applies a prefix list to OSPF for IPv6 routing updates. Only routes permitted by the prefix-list can go into the routing table.

## Syntax

**distribute-list prefix-list** *list-name* **in** [ **ethernet** *unit/slot/port* | **lag** *lag-id* | **loopback** *number* | **tunnel** *number* | **ve** *virtual port number* ]

**no distribute-list prefix-list**

## Command Default

Prefix lists are not applied to OSPFv3 for IPv6 routing updates.

## Parameters

*list-name*
Name of a prefix-list. The list defines which OSPFv3 networks are to be accepted in incoming routing updates.

**in**
Applies the prefix list to incoming routing updates on the specified interface.

**ethernet** *unit/slot/port*
Specifies an Ethernet interface.

**lag** *lag-id*
Specifies a LAG virtual interface.

**loopback** *number*
Specifies a loopback interface and port number.

**tunnel** *number*
Specifies a tunnel.

**ve** *virtual port number*
Specifies a virtual Ethernet (VE) interface.

## Modes

OSPFv3 router configuration mode

OSPFv3 VRF router configuration mode

## Usage Guidelines

The **no** form of the command removes the prefix list.

# Examples

The following example configures a distribution list that applies the filterOspfRoutes prefix list globally.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# distribute-list prefix-list filterOspfRoutes in
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* option. |

# distribute-list prefix-list (RIPng)

Applies a prefix list to RIPng to control routing updates that are received or sent.

## Syntax

**distribute-list prefix-list** *list-name* { **in** | **out** }

**no distribute-list prefix-list** *list-name* { **in** | **out** }

## Command Default

Prefix lists are not applied to RIPng routing updates.

## Parameters

*list-name*

Specifies the prefix list to be applied.

**in**

Applies the prefix list to incoming routing updates.

**out**

Applies the prefix to outgoing routing updates.

## Modes

RIPng router configuration mode.

## Usage Guidelines

Use the **no** form of the command to remove the distribution list.

## Examples

The first prefix list in the following example denies routes with the prefix beginning with 2001:db8:: if the prefix is longer than 64 bits. The second prefix list allows all other routes received.

```
device# configure terminal
device(config)# ipv6 prefix-list 2001routes deny 2001:db8::/64 le 128
device(config)# ipv6 prefix-list 2001routes permit ::/0 ge 0 le 128
device(config)# ipv6 router rip
device(config-ripng-router)# distribute-list prefix-list 2001routes in
```

# distribute-list route-map

Creates a route-map distribution list.

## Syntax

**distribute-list route-map** *map* **in**

**no distribute-list route-map**

## Parameters

*map*

Specifies a route map.

**in**

Creates a distribution list for an inbound route map.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The distribution list can filter Link State Advertisements (LSAs) received from other OSPF devices before adding the corresponding routes to the routing table.

The **no** form of the command removes the distribution list.

## Examples

The following example creates a distribution list using a route map named filter1 that has already been configured.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# distribute-list route-map filter1 in
```

# dlb-internal-trunk-hash

Changes the hashing method for inter-packet-processor (inter-pp) HiGig links that are used to connect master and slave units in ICX 7450-48 devices.

## Syntax

**dlb-internal-trunk-hash** { **inactivity-mode** | **spray-mode** }

**no dlb-internal-trunk-hash** { **inactivity-mode** | **spray-mode** }

## Command Default

The hashing method is inactivity mode.

## Parameters

**inactivity-mode**

Specifies that the flow is set by the inactivity of traffic loading.

**spray-mode**

Specifies that the flow is set to receive new member assignments for every packet arrival in accordance with the traffic loading of each aggregate member.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default hashing method.

**NOTE**
This command is supported only on ICX 7450-48 devices that have master and slave units.

Dynamic load balancing (DLB) enhances hash-based load balancing by taking into account the traffic loading in the network. The inter-pp HiGig links in ICX7450-48 devices use hash-based load balancing to distribute traffic evenly. You can configure the **dlb-internal-trunk-hash** command to change the hashing method.

**NOTE**
Spray mode may introduce out-of-order packet delivery.

## Examples

The following example globally enables spray mode as the inter-pp links hashing method.

```
ICX7450-48P Router(config)#dlb-internal-trunk-hash  spray-mode
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | Added a note about spray mode. |

# dns-filter

Defines Domain Name System (DNS) filters that will restrict DNS queries from unauthenticated hosts to be forwarded explicitly to defined servers.

## Syntax

**dns-filter** *filter-id ip-address wildcard-bits*

**no dns-filter** *filter-id ip-address wildcard-bits*

## Command Default

DNS filters are not defined.

## Parameters

*filter-id*

Defines the number to identify a DNS filter. The valid values are from 1 through 4.

*ip-address*

Specifies the IP address (A.B.C.D) or IP address along with the prefix length (A.B.C.D/n) of unauthenticated hosts.

*wildcard-bits*

Specifies a wildcard for the filter. The wildcard is in dotted-decimal notation (IP address format).

## Modes

Web Authentication configuration mode

## Usage Guidelines

Many of the Web Authentication solutions allow DNS queries to be forwarded from unauthenticated hosts. To eliminate the threat of forwarding DNS queries from unauthenticated hosts to unknown or untrusted servers (also known as domain-casting), you can restrict DNS queries from unauthenticated hosts to be forwarded explicitly to defined servers by defining DNS filters. Any DNS query from an unauthenticated host to a server that is not defined in a DNS filter is dropped. Only DNS queries from unauthenticated hosts are affected by DNS filters; authenticated hosts are not. If the DNS filters are not defined, then any DNS queries can be made to any server.

The wildcard is in dotted-decimal notation (IP address format). It is a four-part value, where each part is 8 bits (one byte) separated by dots, and each bit is a one or a zero. Each part is a number ranging from 0 to 255, for example 0.0.0.255. Zeros in the mask mean the packet source address must match the IP address. Ones mean any value matches. For example, the IP address and subnet-mask values 10.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 10.157.22.x match the policy.

The **no** form of the command removes the defined DNS filters.

# Examples

The following example defines a DNS filter.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# dns-filter 2 192.168.10.1/24 0.0.0.255
```

# domain-name

Configures the domain name for the DHCP client.

## Syntax

**domain-name** *domain-name*

## Parameters

*domain-name*

Specifies the name of the domain.

## Modes

DHCP server pool configuration mode

## Examples

The following example specifies the domain name for the DHCP client.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# domain-name sierra
```

# dot1x auth-filter

Applies the specified filter on the interface and the MAC addresses defined in the filter (MAC filter) do not have to go through authentication.

## Syntax

**dot1x auth-filter** *filter-id vlan-id*

**no dot1x auth-filter** *filter-id vlan-id*

## Command Default

There are no filters applied on the interface.

## Parameters

*filter-id*

Specifies the filter ID to be applied on the interface.

*vlan-id*

Specifies the VLAN ID.

## Modes

Interface configuration mode

## Usage Guidelines

A client can be authenticated in an untagged VLAN or tagged VLAN using the MAC address filter for 802.1X authentication.

If auth-filter has tagged VLAN configuration, the clients are authenticated in auth-default VLAN and tagged VLAN provided in auth-filter. The clients authorized in auth-default VLAN allow both untagged and tagged traffic.

The following rules apply when using the **dot1x auth-filter** command:

- The maximum number of filters that can be bound to a port is limited by the mac-filter-port default or a configured value.
- The filters must be applied as a group. For example, if you want to apply four filters to an interface, they must all appear on the same command line.
- You cannot add or remove individual filters in the group. To add or remove a filter on an interface, apply the filter group again containing all the filters you want to apply to the port.
- If you apply a filter group to a port that already has a filter group applied, the older filter group is replaced by the new filter group.
- If you add filters to or modify the 802.1X authentication filter, the system clears all 802.1X sessions on the port. Consequently, all users that are logged in will need to be re-authenticated.

The **no** form of the command disable the dot1x auth-filter functionality. If the VLAN is not specified, the auth-default-vlan is used.

# Examples

The following example applies the dot1x filter on a specific VLAN.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# dot1x auth-filter 1 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# dot1x enable

Enables 802.1X authentication globally.

## Syntax

**dot1x enable** [ **all** | **ethernet** *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/***port* ] ]

**no dot1x enable** [ **all** | **ethernet** *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ] ]

## Command Default

802.1x authentication is not enabled.

## Parameters

**all**

Enables 802.1x authentication on all interfaces.

**ethernet** *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ]

Enables 802.1x authentication on the specified interface or range of interfaces.

## Modes

Authentication configuration mode

## Usage Guidelines

The **dot1x enable** command without any options initializes 802.1X authentication feature globally. The **dot1x enable** command with the **all** or **ethernet** options, enables 802.1X authentication on all or a specific interface respectively. After initializing 802.1X authentication feature using the **dot1x enable** command, you must enable 802.1X authentication on all or a specific interface.

Port control must be configured to activate authentication on an 802.1X-enabled interface using the **dot1x port-control** command from the interface configuration mode.

The **no** form of the command disables 802.1X authentication.

**NOTE**
You cannot enable 802.1X authentication on ports that have any of the following features enabled:

- Link aggregation
- Metro Ring Protocol (MRP)
- Mirror port
- LAG port
- Unidirectional Link Detection (UDLD)

# Examples

The following example enables 802.1X authentication on all interfaces.

```
device(config)# authentication
device(config-authen)# dot1x enable
device(config-authen)# dot1x enable all
```

The following example enables 802.1X authentication on ethernet interface 1/1/1.

```
device(config)# authentication
device(config-authen)# dot1x enable
device(config-authen)# dot1x enable ethernet 1/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# dot1x guest-vlan

Specifies the VLAN into which the port should be placed when the client's response to the dot1x requests for authentication times out.

## Syntax

**dot1x guest-vlan** *vlan-id*

**no dot1x guest-vlan** *vlan-id*

## Command Default

The guest VLAN ID is not specified.

## Parameters

*vlan-id*

Specifies the VLAN ID of the guest VLAN.

## Modes

dot1x configuration mode.

## Usage Guidelines

The **no** form of this command disables the functionality.

Use this command when the client does not support the 802.1X authentication, so that the client can access default privileges.

If there is no response from dot1x client for EAP-packets and if guest VLAN is not configured, authentication is considered as failed and the configured failure action is performed.

## Examples

The following example specifies the guest VLAN.

```
device(config)# authentication
device(config-authen)# dot1x guest-vlan 7
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# dot1x initialize

Initializes 802.1X authentication on a port.

## Syntax

**dot1x initialize ethernet** *unit/slot/port*

## Parameters

**ethernet** *unit/slot/port*

Specifies the details of the interface on which 802.1x authentication is to be initialized.

## Modes

Privileged EXEC mode

## Examples

The following example initializes dot1x authentication on a port.

```
device# dot1x initialize ethernet 3/1/1
```

# dot1x macauth-override

Sets an override option so that MAC authentication is attempted when 802.1X authentication fails for the client.

## Syntax

**dot1x macauth-override**

**no dot1x macauth-override**

## Command Default

By default, client authentication fails if 802.1X authentication fails.

## Modes

authentication configuration sub-mode.

## Examples

The following example configures **dot1x macauth-override** on a device.

```
device# configure terminal
device(config)# authentication
device(config-authen)# dot1x macauth-override
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# dot1x max-reauth-req

Configure the maximum number of times (attempts) EAP-request/identity frames are sent for reauthentication after the first authentication attempt.

## Syntax

**dot1x max-reauth-req** *count*

**no dot1x max-reauth-req** *count*

## Command Default

The device sends the EAP-request/identity frames for reauthentication twice.

## Parameters

*count*

Specifies the number of EAP frame re-transmissions. This is a number from 1 through 10. The default is 2.

## Modes

Authentication configuration mode

## Usage Guidelines

The **no** form of this command will disable this functionality.

The ICX device retransmits the EAP-request/identity frame a maximum of two times. If no EAP response/identity frame is received from the client after two EAP-request/identity frame re-transmissions (or the amount of time specified with the max-reauth-req command), the device restarts the authentication process with the client.

You can optionally change the number of times the device should retransmit the EAP request/identity frame.

## Examples

The following example configures the device to retransmit an EAP-request/identity frame to a client a maximum of three times.

```
device(config)# authentication
device(config-authen)# dot1x max-reauth-req 3
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# dot1x max-req

Configures the retransmission parameter that defines the maximum number of times EAP request/challenge frames are retransmitted when EAP response/identity frame is not received from the client.

## Syntax

**dot1x max-req** *count*

**no dot1x max-req** *count*

## Command Default

The device retransmits the EAP-request/challenge twice.

## Parameters

*count*

Specifies the number of EAP frame re-transmissions. Th range is from from 1 through 10. The default value is 2.

## Modes

Authentication configuration mode

## Usage Guidelines

The **no** form of the command disables this functionality.

## Examples

The following example configures the device to retransmit an EAP-request/challenge frame to a client a maximum of three times.

```
device(config)# authentication
device(config-authen)# dot1x max-req 3
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# dot1x-mka-enable

Enables MACsec Key Agreement (MKA) capabilities on a licensed device and enters dot1x-mka configuration mode.

## Syntax

**dot1x-mka-enable**

**no dot1x-mka-enable**

## Command Default

No MACsec capability is available.

## Modes

Global configuration

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The **no** form of this command disables the MKA and MACsec functionality on all ports. This may require the already authenticated hosts to re-authenticate.

Use the **dot1x-mka-enable** command to enable MACsec on an already licensed device. Commands may be visible, but they do not work on a non-licensed device.

## Examples

The following example enables MACsec capabilities on the device.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | MACsec support was added on ICX 7650 devices. |

## Related Commands

enable-mka, mka-cfg-group

# dot1x port-control

Controls port-state authorization and configures the port control type to activate authentication on an 802.1X-enabled interface.

## Syntax

**dot1x port-control** { **auto** | **force-authorized** | **force-unauthorized** } { **all** | **ethernet** *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ] }

**no dot1x port-control** { **auto** | **force-authorized** | **force-unauthorized** } { **all** | **ethernet** *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/***port* ] }

## Command Default

All controlled ports on the device are in the authorized state, allowing all traffic.

## Parameters

**auto**

Enables authentication on a port. It places the controlled port in the unauthorized state until authentication takes place between the client and authentication server. Once the client passes authentication, the port becomes authorized. This activates authentication on an 802.1X-enabled interface. The controlled port remains in the authorized state until the Client logs off.

**force-authorized**

Places the controlled port unconditionally in the authorized state, allowing all traffic to pass between the client and the authenticator. This is the default state for ports on the device.

**force-unauthorized**

Places the controlled port unconditionally in the unauthorized state, denying any traffic to pass between the client and the authenticator.

**ethernet** *unit* **/** *slot* **/** *port* [ **to** *unit* **/** *slot* **/** *port* ]

Configures the specified interface or range of interfaces.

**all**

Configures all interfaces on the device.

## Modes

General configuration mode

## Usage Guidelines

Before activating the authentication using the **dot1x port-control auto** command on an untagged port, you must remove configured static ACL, if any, from the port.

You cannot enable 802.1X authentication on ports that have any of the following features enabled:

- Link aggregation

- Metro Ring Protocol (MRP)
- Mirror port
- LAG port

The **no** form of the command resets the port control type to the default state.

# Examples

The following example places the configured port unconditionally in the unauthorized state until authentication takes place between the client and authentication server. Once the client passes authentication, the port becomes authorized.

```
device# configure terminal
device(config)# dot1x port-control auto ethernet 3/1/1
```

The following example configures the specified interface to place the controlled port unconditionally in the authorized state.

```
device# configure terminal
device(config)# dot1x port-control force-authorized ethernet 3/1/1
```

The following example configures the specified interface to place the controlled port unconditionally in the unauthorized state.

```
device# configure terminal
device(config)# dot1x port-control force-unauthorized ethernet 3/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was moved to general configuration level. |

# dot1x timeout

Configures the timeout parameters that determine the time interval for client reauthentication and EAP retransmissions.

## Syntax

**dot1x timeout** {**quiet-period** *seconds*| **supplicant** *seconds* | **tx-period** *seconds* }

**no dot1x timeout** {**quiet-period** *seconds*| **supplicant** *seconds* | **tx-period** *seconds* }

## Command Default

The timeout parameters are not applied to the device.

## Parameters

**quiet-period** *seconds*

Specifies the time, in seconds, the device waits before trying to re-authenticate the client. The quiet period can be from 1 through 4294967295 seconds. The default is 60 seconds. If the Ruckus device is unable to authenticate the client, the ICX device waits a specified amount of time before trying again. The amount of time the device waits is specified with the quiet period parameter.

**supplicant** *seconds*

By default, when the ICX device relays an EAP-Request frame from the RADIUS server to the client, it expects to receive a response from the client within 30 seconds. You can optionally specify the wait interval using the **supplicant** *seconds* parameters. The value is 1 through 4294967295.

**tx-period** *seconds*

Specifies the EAP request retransmission interval, in seconds, with the client. By default, if the device does not receive an EAP-response/identity frame from a client, the device waits 30 seconds, then retransmits the EAP-request/identity frame. You can optionally change the amount of time the device waits before re-transmitting the EAP-request/identity frame to the client. If the client does not send back an EAP-response/identity frame within 60 seconds, the device will transmit another EAP-request/identity frame. The tx-period is a value from 1 through 4294967295. The default is 30 seconds.

## Modes

Authentication configuration mode

## Usage Guidelines

The **no** form of the command disables dot1x timeout.

## Examples

The following example specifies the quiet period as 30 seconds.

```
device(config)# authentication
device(config-authen)# dot1x enable
device(config-authen)# dot1x timeout quiet-period 30
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# dynamic

Configures dynamic ports.

## Syntax

**dynamic**

**no dynamic**

## Command Default

Ports are static.

## Modes

Protocol VLAN configuration mode

## Usage Guidelines

Dynamic ports within any protocol VLAN age out after 10 minutes if no member protocol traffic is received on a port within the VLAN. Once you dynamically add a port to a protocol VLAN, you cannot configure routing parameters on the port. You cannot dynamically add a port to a protocol VLAN if the port has any routing configuration parameters.

**NOTE**
Dynamic addition and removal of ports is not applicable for an AppleTalk protocol VLAN. You cannot route to or from protocol VLANs with dynamically added ports. In the switch image, all the ports are dynamic ports by default.

The **no** form of the command removes the dynamic setting.

## Examples

The following example shows the IP protocol VLAN configured with dynamic ports.

```
device(config)# vlan 10
device(config-vlan-10)# ip-proto name IP_Prot_VLAN
device(config-vlan-ip-proto)# dynamic
```

The following example shows configuring port-based VLAN 10, and then configuring an IP subnet VLAN within the port-based VLAN with dynamic ports.

```
device(config)# vlan 10 name IP_VLAN by port
device(config-vlan-10)# untagged ethernet 1/1/1 to 1/1/6
added untagged port ethernet 1/1/1 to 1/1/6 to port-vlan 10.
device(config-vlan-10)# ip-subnet 10.1.1.0/24 name Mktg-LAN
device(config-vlan-ip-subnet)# dynamic
```

The following example shows configuring port-based VLAN 20, and then configuring an IPX network VLAN within the port-based VLAN with dynamic ports. These commands create a port-based VLAN on chassis ports 1/2/1 through 1/2/6 named "Eng-LAN", configure an IPX network VLAN within the port-based VLAN, and then add ports from the port-based VLAN dynamically.

```
device(config)# vlan 20 name IPX_VLAN by port
device(config-vlan-10)# untagged ethernet 1/2/1 to 1/2/6
added untagged port ethernet 1/2/1 to 1/2/6 to port-vlan 20.
device(config-vlan-10)# ipx-network abcd ethernet_ii name Eng-LAN
device(config-vlan-ipx-network)# dynamic
```

# eckeypair (PKI)

Specifies which EC keypair to use during enrollment.

## Syntax

**eckeypair** { **key-label** *keyname* }

**no eckeypair** { **key-label** *keyname* }

## Command Default

## Parameters

**key-label**

Precedes the keyname to be used for enrollment.

*keyname*

Specifies the name of the pre-existing key to be used for enrollment.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the eckeypair from enrollment configuration.

## Examples

The following example creates a trustpoint named trust1 and configures it to use the ec keypair eckeyAuto.

```
device(config)# pki trustpoint trust1
device(config-pki-trustpoint-trust1)# auto-enroll
device(config-pki-trustpoint-trust1)# enrollment retry-period 2
device(config-pki-trustpoint-trust1)# enrollment profile profile1
device(config-pki-trustpoint-trust1)# pki-entity entity1
device(config-pki-trustpoint-trust1)# eckeypair key-label eckeyAuto
device(config-pki-trustpoint-trust1)# fingerprint 36:0c:92:6e:df:b2:72:eb:59:e8:63:73:2a:98:a8:91:cb:
50:94:d9
device(config-pki-trustpoint-trust1)# ocsp http post
device(config-pki-trustpoint-trust1)# exit
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# eee

Enables Energy Efficient Ethernet (EEE) globally, per port or on a range of ports.

## Syntax

**eee**

**no eee**

## Command Default

Energy Efficient Ethernet is not enabled.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables Energy Efficient Ethernet.

## Examples

The following example enables Energy Efficient Ethernet globally.

```
device(config)# eee
EEE Feature Enabled
```

The following example enables Energy Efficient Ethernet on multiple ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/12
device(config-mif-1/1/1-1/1/12)# eee
EEE Feature Enabled
EEE Feature Enabled on port 1/1/1
EEE Feature Enabled on port 1/1/2
EEE Feature Enabled on port 1/1/3
EEE Feature Enabled on port 1/1/4
EEE Feature Enabled on port 1/1/5
EEE Feature Enabled on port 1/1/6
EEE Feature Enabled on port 1/1/7
EEE Feature Enabled on port 1/1/8
EEE Feature Enabled on port 1/1/9
EEE Feature Enabled on port 1/1/10
EEE Feature Enabled on port 1/1/11
EEE Feature Enabled on port 1/1/12
```

The following example enables Energy Efficient Ethernet per port.

```
device(config)# interface ethernet e1000-1/1/1
device(config-if-e1000-1/1/1)# eee
EEE Feature Enabled EEE on port 1/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# egress-buffer-profile

Attaches a user-configured egress buffer profile to one or more ports.

## Syntax

**egress-buffer-profile** *profile-name*

**no egress-buffer-profile** *profile-name*

## Command Default

If a port is not attached to a user-configured egress buffer profile, it uses the default egress buffer profile.

## Parameters

*profile-name*

Specifies the name of the egress buffer profile to be attached to the port.

## Modes

Interface mode

Multiple-interface mode

## Usage Guidelines

The **no** form of this command removes a user-configured egress buffer profile from the port and the port uses the default egress buffer profile.

You must configure an egress buffer profile before you can attach it to a port.

Only one egress buffer profile at a time can be attached to any port. You can attach an egress buffer profile to more than one port.

## Examples

The following example attaches an egress buffer profile named egress1 to a port:

```
Device(config-if-e10000-1/1/1)# egress-buffer-profile egress1
```

The following example attaches an egress buffer profile named egress2 to multiple ports:

```
Device(config-mif-1/1/2-1/1/16)# egress-buffer-profile egress2
```

The following example removes an egress buffer profile named egress2 from multiple ports:

```
Device(config-mif-1/1/2-1/1/16)# no egress-buffer-profile egress2
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# enable (GVRP)

Enables GVRP on ports.

## Syntax

**enable** { **all** | **ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ] }

**no enable** { **all** | **ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ] }

## Command Default

GVRP is not enabled on ports.

## Parameters

**all**

Enables GVRP on all ports.

**ethernet** *stackid/slot/port*

Enables GVRP on the specified port.

**to** *stackid/slot/port*

Specifies the range of ports upon which to enable GVRP.

## Modes

GVRP configuration mode

## Usage Guidelines

The **no** form of the command disables GVRP.

## Examples

The following example shows how to enable GVRP on all ports.

```
device(config)# gvrp-enable
device(config-gvrp)# enable all
```

The following example shows how to enable GVRP on a list of specific ports.

```
device(config)# gvrp-enable
device(config-gvrp)# enable ethernet 1/1/24 ethernet 1/2/24 ethernet 1/4/17
```

The following example shows how to enable GVRP on a range of ports.

```
device(config)# gvrp-enable
device(config-gvrp)# enable ethernet 1/1/1 to 1/1/8
```

The following example shows how to enable GVRP on a range of ports and a list of ports.

```
device(config)# gvrp-enable
device(config-gvrp)# enable ethernet 1/1/1 to 1/1/8 ethernet 1/1/24 ethernet 1/6/24 ethernet 1/8/17
```

# enable (LAG)

Enables an individual port within a LAG.

## Syntax

**enable port-name** *name*

**enable ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ]

## Command Default

Ports within a LAG are not enabled.

## Parameters

**port-name** *name*
Enables a named port within a LAG.

**ethernet** *unit/slot/port*
Enables the specified Ethernet port within the LAG.

**to** *unit/slot/port*
Enables a range of ports within the LAG.

**lag** *lag-id*
Enables the LAG virtual interface.

## Modes

LAG configuration mode

## Usage Guidelines

To enable a port belonging to a keep-alive LAG, you must use the **enable** command from the interface configuration mode.

## Examples

The following example shows how to enable a port within a LAG configuration.

```
device(config)# lag blue static id 1
device(config-lag-blue)# ports ethernet 1/1/1 ethernet 1/1/5
device(config-lag-blue)# enable ethernet 1/3/1
```

The following example shows how to enable a port belonging to a keep-alive LAG.

```
device(config)# lag test keep-alive
device(config-lag-test)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# enable (MAC Port Security)

Enables MAC port security.

## Syntax

**enable**

**no enable**

## Command Default

By default, MAC port security is disabled on all interfaces.

## Modes

Port security configuration mode

Port security interface configuration mode

## Usage Guidelines

The **no** form of the command disables the MAC port security.

## Examples

The following example enables MAC port security on all interfaces.

```
device(config)# port security
device(config-port-security)# enable
```

The following example enables MAC port security on a specific interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port security
device(config-port-security-e1000-1/1/1)# enable
```

# enable (MRP)

Enables the metro ring.

## Syntax

**enable**

**no enable**

## Command Default

The metro ring is disabled by default.

## Modes

Metro ring configuration mode

## Usage Guidelines

The **no** form of the command disables the metro ring.

## Examples

The following example enables the metro ring.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-1)# enable
```

# enable (Port)

Enables a port.

## Syntax

**enable**

## Command Default

A port is enabled (active).

## Modes

Interface configuration mode

## Usage Guidelines

A port can be deactivated (disabled) or activated (enabled) by selecting the appropriate status.

## Examples

The following example enables a disabled port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# enable
```

# enable (VSRP)

Enables the VSRP VRID for a port-based VLAN.

## Syntax

**enable**

**disable**

## Command Default

The VSRP VRID is disabled by default.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The device must be set as a backup. Because VSRP does not have an owner, all VSRP devices are backups. The active device for a VRID is elected based on the VRID priority.

The **disable** command deactivates VSRP.

## Examples

The following example shows how to enable the VSRP VRID on a VLAN.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# backup
device(config-vlan-200-vrid-1)# enable
```

# enable (Web Authentication)

Enables Web Authentication.

## Syntax

**enable**

**no enable**

## Command Default

Web Authentication is disabled.

## Modes

Web Authentication configuration mode

## Usage Guidelines

The **no** form of the command disables Web Authentication.

## Examples

The following example enables Web Authentication on VLAN 10.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# enable
```

# enable aaa console

Enables AAA support for commands entered at the console.

## Syntax

**enable aaa console**

**no enable aaa console**

## Command Default

Command authorization and command accounting for console commands are not enabled.

## Modes

Global configuration mode

## Usage Guidelines

The ICX device supports command authorization and command accounting for CLI commands entered at the console.

AAA support for commands entered at the console includes the following:

- The login prompt that uses AAA authentication, using authentication method lists
- EXEC authorization
- EXEC accounting
- Command authorization
- Command accounting
- System accounting

The **no** form of the command disables the support for AAA commands entered at the console.

**NOTE**
If you have previously configured the device to perform command authorization using a RADIUS server, entering the **enable aaa console** command may prevent the execution of any subsequent commands entered on the console. This happens because RADIUS command authorization requires a list of allowable commands from the RADIUS server. This list is obtained during RADIUS authentication. For console sessions, RADIUS authentication is performed only if you have configured **aaa authentication enable** and specified RADIUS as the authentication method (for example, with the **aaa authentication enable default radius** command). If RADIUS authentication is never performed, the list of allowable commands is never obtained from the RADIUS server. Consequently, there would be no allowable commands on the console.

If this command is configured, the DHCP client does not request the configuration files as part of the DHCP client auto-provisioning process. Refer to the *Ruckus FastIron DHCP Configuration Guide* for more information.

# Examples

The following example shows how to configure command authorization and command accounting for console commands.

```
device(config)# enable aaa console
```

# enable acl-per-port-per-vlan

Enables support for access control list (ACL) filtering based on VLAN membership or virtual interface (VE) port membership.

## Syntax

**enable acl-per-port-per-vlan**

**no enable acl-per-port-per-vlan**

## Command Default

ACL filtering based on VLAN membership or VE port membership is disabled.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only for IPv4 ACLs, and only for inbound traffic.

This command must be followed by the **write-memory** and **reload** commands to place the change into effect.

IPv4 ACLs that filter based on VLAN membership or VE port membership (ACL per port per VLAN), are supported together with IPv6 ACLs on the same device, as long as they are not bound to the same port or virtual interface.

For DHCPv6 snooping, enter the **enable acl-per-port-per-vlan** command and enable DHCPv6 snooping on both client and server VLANs.

The **no** form of the command disables support for ACL filtering based on VLAN membership or VE port membership.

## Examples

The following example enables support for ACL filtering based on VLAN membership or VE port membership.

```
device(config)# enable acl-per-port-per-vlan
device(config)# write memory
device(config)# exit
device# reload
```

# enable egress-acl-on-cpu-traffic

Enables applying outbound ACLs to traffic generated by the CPU.

## Syntax

**enable egress-acl-on-cpu-traffic**

**no enable egress-acl-on-cpu-traffic**

## Command Default

By default, outbound ACLs are not applied to traffic generated by the CPU.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets to the default; that is, outbound ACLs are not applied to traffic generated by the CPU.

## Examples

The following example shows how to apply outbound ACLs to traffic generated by the CPU.

```
device# configure terminal
device(config)# enable egress-acl-on-cpu-traffic
```

# enable nd hop-limit

For an IPv6 ACL, enables dropping neighbor discovery (ND) packets for which the hop limit is less than 255.

## Syntax

**enable nd hop-limit**

**no enable nd hop-limit**

## Command Default

Hop-limit check for neighbor discovery (ND) packets is disabled.

## Modes

IPv6 ACL configuration mode

## Usage Guidelines

Checking for ND packets with hop limit less than 255 helps protect the device from denial of service (DoS) attacks.

ACLs enabled for hop-limit check are effective only when applied to interfaces. (If you configure an ACL that is already applied to an interface, there is no need to re-apply it.)

This command is effective in ACLs applied to all types of supported interface—physical, port-channel, and VE.

This command applies to the following types of ND packets:

- neighbor advertisement (NA)
- neighbor solicitation (NS)
- router advertisement (RA)
- router solicitation (RS)

To disable hop-limit check for ND packets, use the **no** form of this command.

## Examples

The following example enables hop-limit check for the IPv6 ACL being configured.

```
device# configure terminal
device(config)# ipv6 access-list hl_acl
device(config-ipv6-access-list hl_acl)# enable nd hop-limit
```

The following example disables hop-limit check for the IPv6 ACL being configured.

```
device# configure terminal
device(config)# ipv6 access-list hl_acl
device(config-ipv6-access-list hl_acl)# no enable nd hop-limit
```

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |
| 08.0.30p | Support for this command for ICX 7000 series devices was added. |

# enable password-display

Enables the display of the community string.

## Syntax

**enable password-display**

**no enable password-display**

## Command Default

The display of the community string is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **enable password-display** command enables display of the community string in the output of the **show snmp server** command. Display of the community string remains encrypted in the startup-config and running-config files. When the **enable password-display** command is configured, the user password and SNMP community string are encrypted in the **show run** command output.

The **no** form of the command disables the display of the community string in the output of the **show snmp server** command.

## Examples

The following example shows how to enable the display of the community string.

```
device(config)# enable password-display
```

# enable password-min-length

Configures the minimum length on the Line (Telnet), Enable, or Local passwords.

## Syntax

**enable password-min-length** *length*

**no enable password-min-length** *length*

## Command Default

The password length is one character.

## Parameters

*length*

The number of characters or the length of the password. The range is from 1 through 48. The default is 1.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the password length to the default.

## Examples

The following example shows how to specify that the Line, Enable, and Local passwords be at least 8 characters.

```
device(config)# enable password-min-length 8
```

# enable port-config-password

Allows read-and-write access for specific ports but not for global (systemwide) parameters.

## Syntax

**enable port-config-password** [ *password* ]

**no enable port-config-password** [ *password* ]

## Command Default

Read-write access for specific ports is not configured.

## Parameters

*password*

Alphanumeric password string.

## Modes

Global configuration mode

## Usage Guidelines

You can set one password for each of the management privilege levels: Super User level, Port Configuration level, and Read Only level.

You also can configure up to 16 user accounts consisting of a username and password, and assign each user account to one of the three privilege levels.

**NOTE**
You must set the Super User level password before you can set other types of passwords.

**NOTE**
You must use the CLI to assign a password for management privilege levels. You cannot assign a password using the Web Management Interface.

The **no** form of the command removes the configured password access.

## Examples

The following example shows how to set Port Configuration level password.

```
device(config)# enable port-config-password password1
```

# enable read-only-password

Allows access to the Privileged EXEC mode and User EXEC mode of the CLI, but only with read access.

## Syntax

**enable read-only-password** [ *password* ]

**no enable read-only-password** [ *password* ]

## Command Default

Read access for the Privileged EXEC and User EXEC modes of the CLI is not configured.

## Parameters

*password*

Alphanumeric password string.

## Modes

Global configuration mode

## Usage Guidelines

You can set one password for each of the management privilege levels: Super User level, Port Configuration level, and Read Only level.

You also can configure up to 16 user accounts consisting of a username and password, and assign each user account to one of the three privilege levels.

**NOTE**
You must set the Super User level password before you can set other types of passwords.

**NOTE**
You must use the CLI to assign a password for management privilege levels. You cannot assign a password using the Web Management Interface.

The **no** form of the command removes the configured password access.

## Examples

The following example shows how to set Read Only level password.

```
device(config)# enable read-only-password password1
```

# enable snmp

Enables SNMP access modes.

## Syntax

**enable snmp** { **config-tacacs** | **config-radius** | **ve-statistics** }

**no enable snmp** { **config-tacacs** | **config-radius** | **ve-statistics** }

## Command Default

The SNMP access modes for TACACS and RADIUS are disabled.

## Parameters

**config-tacacs**
Enables TACACS configuration access mode.

**config-radius**
Enables RADIUS configuration access mode.

**ve-statistics**
Enables the display of virtual port statistics.

## Modes

Global configuration mode

## Usage Guidelines

To configure TACACS, TACACS+ or RADIUS authentication parameters, you must enable the corresponding SNMP access mode.

The **no** form of the command disables the SNMP access modes.

## Examples

The following example shows how to enable the SNMP access mode for TACACS.

```
device(config)# enable snmp config-tacacs
```

The following example shows how to enable the SNMP access mode for RADIUS.

```
device(config)# enable snmp config-radius
```

The following example shows how to enable the display of virtual port statistics.

```
device(config)# enable snmp ve-statistics
```

# enable strict-password-enforcement

Enables the password security feature.

## Syntax

**enable strict-password-enforcement**

**no enable strict-password-enforcement**

## Command Default

Strict password is not enforced.

## Modes

Global configuration mode

## Usage Guidelines

When strict password enforcement is enabled on the ICX device, you must enter a minimum of eight characters containing the following combinations when you create an enable and a user password:

- At least two uppercase characters
- At least two lowercase characters
- At least two numeric characters
- At least two special characters

**NOTE**
Password minimum character and combination requirements are strictly enforced.

Passwords must not share four or more concurrent characters with any other password configured on the router. If the you try to create a password with four or more concurrent characters, an error message will be returned.

If you try to configure a password that was previously used, the Local User Account configuration will not be allowed and an error message will be displayed.

The **no** form of the command disables strict password enforcement.

## Examples

The following example shows how to enable strict password enforcement.

```
device(config)# enable strict-password-enforcement
```

# enable super-user-password

Allows complete read-and-write access to the system.

## Syntax

**enable super-user-password** [ *password* ]

**no enable super-user-password** [ *password* ]

## Command Default

Complete read-write access to the system is not configured.

## Parameters

*password*

Alphanumeric password string.

## Modes

Global configuration mode

## Usage Guidelines

You can set one password for each of the management privilege levels: Super User level, Port Configuration level, and Read Only level. The **enable super-user-password** command is generally for system administrators only. The Super User privilege level allows you to configure passwords.

You also can configure up to 16 user accounts consisting of a username and password, and assign each user account to one of the three privilege levels.

You must set the Super User level password before you can set other types of passwords.

**NOTE**
You must use the CLI to assign a password for management privilege levels. You cannot assign a password using the Web Management Interface.

The **no** form of the command removes the configured password access.

## Examples

The following example shows how to set the Super User level password.

```
device(config)# enable super-user-password password1
```

# enable telnet

Configures Telnet access control parameters.

## Syntax

**enable telnet** { **authentication** | **password** *password* }

**no enable telnet** { **authentication** | **password** *password* }

## Command Default

Telnet authentication is not enabled and the Telnet password is not set.

## Parameters

**authentication**
Enables Telnet authentication.

**password** *password*
Sets a password for Telnet access.

## Modes

Global configuration mode

## Usage Guidelines

To authenticate Telnet access to the CLI, you also must enable the authentication by entering the **enable telnet authentication** command. You cannot enable Telnet authentication using the Web Management Interface.

The **no** form of the command removes the Telnet authentication or Telnet password.

## Examples

The following example shows how to enable Telnet authentication.

```
device(config)# enable telnet authentication
```

The following example shows how to set the password for Telnet access.

```
device(config)# enable telnet password pass1
```

# enable user

Configures login and password parameters specific to a user.

## Syntax

**enable user** { **disable-on-login-failure** [ *invalid-attempts* **login-recovery-time** { **in-hours** | **in-mins** | **in-secs** } *recovery-time* ] | **password-aging** | **password-history** [ *previous-passwords* ] | **password-masking** }

**no enable user** { **disable-on-login-failure** [ *invalid-attempts* **login-recovery-time** { **in-hours** | **in-mins** | **in-secs** } *recovery-time* ] | **password-aging** | **password-history** [ *previous-passwords* ] | **password-masking** }

## Command Default

Three login attempts are allowed.

Three minutes of recovery time is enforced before re-enabling user accounts.

In CC mode, the default recovery time is 3 seconds.

The ICX device stores the last five user passwords for each user.

## Parameters

**disable-on-login-failure** *invalid-attempts*

Specifies the number of login attempts before a user is locked out (disabled). The range is from 1 through 10. The default is 3.

**login-recovery-time** { **in-hours** | **in-mins** | **in-secs** } *recovery-time*

Specifies the recovery time in designated units (hours, minutes, or seconds) after which the locked-out user accounts are re-enabled automatically. The valid range for **in-hours** is 1 through 2. The valid range for **in-minutes** is 3 through 120. The valid range for **in-seconds** is 2 through 7200.

**password-aging**

Enables password aging.

**password-history** *previous-passwords*

Specifies how many previous passwords should be stored. The range is from 1 through 15. The default is 5.

**password-masking**

Enables password masking.

## Modes

Global configuration mode

## Usage Guidelines

When password masking is enabled, the CLI displays an asterisk (*) on the console instead of the actual password character entered.

When password aging is enabled, the software records the system time that each user password was configured or last changed. After 180 days, the CLI automatically prompts users to change their passwords when they attempt to sign on. The time displays in the output of the **show running configuration** command, indicated by set-time.

When changing a user password, the user cannot use any of the five previously configured passwords. You can configure the ICX device to store up to 15 passwords for each user, so that users do not use the same password multiple times. If a user attempts to use a password that is stored, the system prompts the user to choose a different password.

If a user fails to log in after three attempts, that user is locked out. You can increase or decrease the number of login attempts before the user is locked-out.

The **no** form of the command removes the login and password configurations.

The **no** form of **enable user disable-on-login-failure** disables both the maximum number of login attempts and recovery time configurations. To disable only the recovery time configuration, use the **no enable user** { **disable-on-login-failure** [ *invalid-attempts* **login-recovery-time** *recovery-time* ] } command.

# Examples

The following example sets the number of login attempts for a user to 10.

```
device(config)# enable user disable-on-login-failure 10
```

The following example configures the user account to automatically re-enable the locked-out users after 5 minutes of the lockout.

```
device(config)# enable user disable-on-login-failure 4 login-recovery-time in-mins 5
```

The following example shows enables password aging.

```
device(config)# enable user password-aging
```

The following example enables password masking. The following example shows how the CLI displays an asterisk (*) on the console instead of the actual password character entered.

```
device(config)# enable user password-masking

device(config)# username xyz password
Enter Password: ********
```

The following example configures the device to store up to 10 previous passwords.

```
device(config)# enable user password-history 10
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | The command was modified to include **login-recovery-time** *recovery-time* option was introduced. |
| 08.0.70 | The command was modified to specify *recovery-time* in hours, minutes, or seconds. The default recovery-time in CC mode was changed to 3 seconds. |

# enable-accounting

Enables Access Control List (ACL) accounting for IPv4 and IPv6 ACLs.

## Syntax

**enable-accounting**

**no enable-accounting**

## Command Default

This option is disabled.

## Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

## Usage Guidelines

The **no** form of this command disables ACL accounting on the associated ACL interface.

## Examples

The following example enables IPv6 ACL accounting. The named access-list must be configured before enabling the ACL accounting.

```
device(config)# ipv6 access-list v6
device(config-ipv6-access-list-v6)# enable-accounting
```

The following example enables ACL accounting for an IPv4 named ACL.

```
device(config)# ip access-list standard std
device(config-std-nacl)# permit 10.10.10.0/24
device(config-std-nacl)# deny 10.20.20.0/24
device(config-std-nacl)# enable-accounting
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# enable-mka

Enables MACsec Key Agreement (MKA) to support MACSec licensing functionality on a specified interface, and changes the mode to dot1x-mka-interface mode to enable related parameters to be configured.

## Syntax

**enable-mka ethernet** *device*/*slot*/*port*

**no enable-mka ethernet** *device*/*slot*/*port*

## Command Default

MKA is not enabled on an interface.

## Parameters

**ethernet** *device*/*slot*/*port*

Specifies an Ethernet interface and the number of the device, the slot on the device, and the port on that slot.

## Modes

dot1x-mka-interface mode

## Usage Guidelines

When the **no** version of the command is executed, MACSec is removed from the port.

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

For a MACsec channel to be created between two ports, both ports and devices designated must have MACsec enabled and configured.

The **enable-mka ethernet** command enables MACSec licensing on the specified interface. If the command is not enabled, MACSec licensing functionality is not supported.

## Examples

The following example enables MACsec on port 2, slot 3 of the first device in the stack.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# enable-mka ethernet 1/3/2
device(config-dot1x-mka-1/3/2)#
```

The following error message is displayed when the MACSec license is not purchased for the device.

```
device(config)# dot1x-mka-enable
device (config-dot1x-mka)# enable-mka ethernet 2/2/1
Error: No MACsec License available for the port 2/2/1. Cannot enable MACsec !!!
Error: MKA cannot be enabled on port 2/2/1
device(config-dot1x-mka)#
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Adds MACsec support for ICX 7650 devices. |

# encapsulation-mode

Specifies the encapsulation mode for an IPsec proposal.

## Syntax

**encapsulation-mode** *encapsulation-mode*

## Command Default

The default encapsulation mode is tunnel mode.

## Parameters

*encapsulation-mode*
Specifies the encapsulation mode. Only tunnel mode is currently supported.

## Modes

IPsec proposal configuration mode

## Usage Guidelines

Because tunnel mode is configured by default and is the only mode that is currently supported, you do not need to configure the encapsulation mode for an IPsec proposal.

## Examples

The following example shows how to configure tunnel mode as the encapsulation mode for an IPsec proposal named ipsec_proposal.

```
device(config)# ipsec proposal ipsec_proposal
device(config-ipsec-proposal-ipsec_proposal)# encapsulation-mode tunnel
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# encryption

Configures an encryption algorithm for an Internet Key Exchange version 2 (IKEv2) proposal.

## Syntax

**encryption** { **aes-cbc-128** | **aes-cbc-256** }

**no encryption** { **aes-cbc-128** | **aes-cbc-256** }

## Command Default

The default encryption algorithm is AES-CBC-256.

## Parameters

**aes-cbc-128**
Specifies the 128-bit advanced encryption standard algorithm in cipher block chaining mode.

**aes-cbc-256**
Specifies the 256-bit advanced encryption standard algorithm in cipher block chaining mode.

## Modes

IKEv2 proposal configuration mode

## Usage Guidelines

Multiple encryption algorithms may be configured for an IKEv2 proposal.

When only one encryption algorithm is configured for an IKEv2 proposal, removing it restores the default configuration.

The **no** form of the command removes the specified encryption algorithm configuration.

## Examples

The following example shows how to configure the AES-CBC-128 encryption algorithm for an IKEv2 proposal named ikev2_proposal.

```
device(config)# ikev2 proposal ikev2_proposal
device(config-ikev2-proposal-ikev2_proposal)# encryption aes-cbc-128
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# encryption-algorithm

Configures an encryption algorithm to protect data traffic for an IPsec proposal.

## Syntax

**encryption-algorithm** { **aes-gcm-256** | **aes-gcm-128** }

**no encryption-algorithm** { **aes-gcm-256** | **aes-gcm-128** }

## Command Default

The default encryption algorithm for an IPsec proposal is AES-GCM-256.

## Parameters

**aes-gcm-256**

Specifies that the 256-bit advanced encryption standard algorithm in Galois counter mode is supported for Encapsulating Security Payload (ESP) encryption.

**aes-gcm-128**

Specifies that the 128-bit advanced encryption standard algorithm in Galois counter mode is supported for ESP encryption.

## Modes

IPsec proposal configuration mode

## Usage Guidelines

Multiple encryption algorithms may be configured for an IPsec proposal.

For an IPsec tunnel to come up successfully, IPsec peer devices must be configured with a common encryption algorithm.

Ruckus ICX 7450 supports dual mode for encryption and decryption. Dual mode is set when both the AES-GCM-128 and AES-GCM-256 algorithms are set for the same IPSec proposal (no further configuration is needed to establish dual mode).

When dual mode is configured on both the local and remote peers, AES-GCM-256 is automatically selected for encryption and decryption.

When dual mode is not configured on both the local and remote peers, the algorithm that is configured on both peers is automatically selected for encryption and decryption.

When only one encryption algorithm is configured for an IPsec proposal, removing it restores the default configuration.

The **no** form of the command removes the specified encryption algorithm configuration.

# Examples

The following example shows how to configure the AES-GCM-128 encryption algorithm for an IPsec proposal named ipsec_prop.

```
device(config)# ipsec proposal ipsec_prop
device(config-ipsec-proposal-ipsec_prop)# encryption-algorithm aes-gcm-128
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# enforce-first-as

Enforces the use of the first autonomous system (AS) path for external BGP (eBGP) routes.

## Syntax

**enforce-first-as**

**no enforce-first-as**

## Modes

BGP configuration mode

## Usage Guidelines

This command causes the router to discard updates received from eBGP peers that do not list their AS number as the first AS path segment in the AS_PATH attribute of the incoming route.

The **no** form of the command disables this feature.

## Examples

The following example configures the device to enforce the use of the first AS path.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# enforce-first-as
```

# enrollment (PKI)

Sets the enrollment retry count, retry period, or profile.

## Syntax

**enrollment** { **retry-count** *count* | **retry-period** *period* | **profile** *profile-name* }

**no enrollment** { **retry-count** *count* | **retry-period** *period* | **profile** *profile-name* }

## Command Default

## Parameters

**retry-count** *count*

Sets the number of enrollment attempts allowed before the next retry-period expires.

**retry-period** *period*

Defines in minutes the wait required before another attempt to enroll after the maximum retry count has been reached. The valid range is 1 through 60 minutes.

**profile** *profile-name*

Designates the profile to be used for enrollment.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

## Examples

The following example sets the allowable enrollment retries to 3, sets a period of 2 minutes for the retry period, and configures the enrollment profile as profile1.

```
device# configure terminal
device(config)# pki trustpoint trust1
device(config-pki-trustpoint-trust1)# enrollment retry-count 3
device(config-pki-trustpoint-trust1)# enrollment retry-period 2
device(config-pki-trustpoint-trust1)# enrollment profile profile1
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# erase system factory-default

Wipes out the system settings and restore factory default settings.

## Syntax

**erase system factory-default**

## Modes

Global configuration mode

## Usage Guidelines

Autocomplete is disabled for this CLI. You must manually key in the command. To prevent accidental execution of this command there are two level of acceptance required from user. First, a warning message is displayed asking user for confirmation. Upon confirmation, a detailed warning message is displayed, prompting the user about various files which may be erased upon executing the command. The factory reset action is triggered, when you accept the warning. The switch will then detect the factory reset action and start performing the below mentioned steps:

- If external USB plugged in, software will unmount it.
- System will be set to same state as shipping out of factory i.e
  - Erase FI config
  - Erase boot config
  - Erase core files
  - Erase sys logs
  - Erase license Info
  - Erase license persistent
  - Erase license files (for XML license file)
- System will auto reboot
- The switch boots up with factory default settings. SAU license is restored to original license info present in original SKU info.

# Examples

```
device# erase system factory-default
"System will go for a reload. Please enter "y" to confirm "n" to exit"
Y
************************************************************
* Factory Reset Alert *
************************************************************
* Please pay attention to the details listed below *
* 1. uboot params will be erased, you might want to *
* backup the uboot params. *
* stop at uboot and do 'printenv' to read uboot params *
* 2. All configuration will be erased, you might want to *
* backup the running config (show running-config) *
* 3. Core Files, Logs will be erased *
* 4. SAU license is restored to original sku *
* use show license sau for more detials *
* 5. XML license is erased
* 6. System will go for a reload *
************************************************************
************************************************************
I have read the alert and factory reset can be performed now
Please enter 'y' to confirm, 'n' to exit :
************************************************************
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# erase flash

Erases an image stored in the system flash.

## Syntax

**erase flash** { **primpary** | **secondary** | **unit-id-pri** *string* | **unit-id-sec** *string* }

## Parameters

**primary**
Erases the primary code image.

**secondary**
Erases the secondary code image.

**unit-id-pri** *string*
Erases the primary code image from the specified stack members. You can specify **all** or a member listwithout blank spaces (2,3,5-7).

**unit-id-sec** *string*
Erases the secondary code image from the specified stack members. You can specify **all** or a member list without blank spaces (2,3,5-7).

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to erase the files stored in the primary or secondary flash or on the stack units.

## Examples

The following example erases the image stored in the secondary flash of the system.

```
device# erase flash secondary
```

The following example erases the image stored in the secondary flash of a set of stack units.

```
device# erase flash unit-id-sec 3,4,5-8,9
```

# erase startup-config

Erases the startup configuration.

## Syntax

**erase startup-config** [ **unit-id** *unit-list* ]

## Parameters

**unit-id** *unit-list*

Erases the startup configuration file from the specified stack member. The member list is specified without blank spaces (2,3,5-7).

## Modes

Privileged EXEC mode

## Examples

The following example erases the startup configuration from specified members in a stack.

```
device# erase startup-config unit-id 2,5,7-8,10
```

# errdisable packet-inerror-detect

Enables the device to monitor configured ports for inError packets and defines the sampling time interval in which the number of inError packets is counted.

## Syntax

**errdisable packet-inerror-detect** *sampling-interval*

**no errdisable packet-inerror-detect** *sampling-interval*

## Command Default

There is no monitoring for inError packets on any port of the device.

## Parameters

*sampling-interval*

Specifies the sampling interval in seconds. It can take a value in the inclusive range of 2 through 60 seconds.

## Modes

Global configuration mode

## Usage Guidelines

If the number of inError packets exceeds the configured threshold for two consecutive sampling windows, then the configured port is error-disabled. The **no** form of this command disables this monitoring.

## Examples

The following example sets the sampling interval in which the number of inError packets is counted to three seconds.

```
device(config)# errdisable packet-inerror-detect 3
```

## History

| Release version | Command history |
|---|---|
| 07.3.00g | This command was introduced. |

# errdisable recovery

Enables a port to recover automatically from the error-disabled state.

## Syntax

**errdisable recovery cause** { **all** | *cause* }

**no errdisable recovery cause** { **all** | *cause* }

**errdisable recovery interval** *time*

**no errdisable recovery interval** *time*

## Command Default

The ports in the error-disabled state are not recovered.

## Parameters

**all**

Enables the ports torecover automatically from an error-disabled state caused by reasons such as BPDU guard violation, the number of inError packets exceeding the configured threshold, a loop detection violation, or the reception of a critical event from the remote device in the case of an EFM-OAM interface.

**cause***cause*

Configures the ports to recover from an error-disabled state caused by one of the following reasons:

- **bpduguard**
- **loam-critical-event**
- **loop-detection**
- **packet-inerror-detect**
- **pvstplus-protect**

**bpduguard**

Configures the port to recover from the error-disabled state if the state was caused because of BPDU guard violation.

**loam-critical-event**

Configures the EFM-OAM interface to recover from the error-disabled state if the state was caused due to reception of a critical event from the remote device.

**loop-detection**

Configures the port to recover from the error-disabled state if the state was caused because of loop detection.

**packet-inerror-detect**

Configures the port to recover from the error-disabled state if the state was caused because the number of inError packets exceeded the configured threshold.

**pvstplus-protect**

Configures the port to recover from the error-disabled state if the state was caused because the PVST+ Protect feature is enabled.

**interval**

Configures a timeout value for the recovery mechanism when the port is in an error-disabled state. Upon the expiry of the timeout value, the ports are automatically recovered.

*time*

Specifies the recovery time interval in seconds for the device to wait before automatically recovering the ports. Range is from 10 through 65535 seconds. The default recovery timeout value is 300 seconds.

# Modes

Global configuration mode

# Usage Guidelines

When automatic recovery re-enables the port, the port is not in the error-disabled state, but it can remain down for other reasons, such as the Tx/Rx of the fibre optic not being seated properly. Thus, the port is not able to receive the signal from the other side. In this case, after the optic is inserted correctly, you must manually disable the port and then re-enable it.

The **no** form of the **errdisable recovery cause** command disables the error-disabled recover functionality.

The **no** form of the **errdisable recovery interval** command reverts to the default recovery time interval value.

# Examples

The following example configures the device to recover the port from the error-disabled state caused because of BPDU guard violation.

```
device(config)# errdisable recovery cause bpduguard
```

The following example configures the device to recover the EFM-OAM interface from the error-disabled state caused by reception of a critical event from the remote device.

```
device(config)# errdisable recovery cause loam-critical-event
```

The following example configures the device to recover the port from the error-disabled state caused because of loop detection.

```
device(config)# errdisable recovery cause loop-detection
```

The following example configures the device to recover the port from the error-disabled state caused because the number of inError packets exceeded the configured threshold.

```
device(config)# errdisable recovery cause packet-inerror-detect
```

The following example configures the device to recover the port from the error-disabled state caused because PVST+ Protect was enabled.

```
device(config)# errdisable recovery cause pvstplus-protect
```

The following example configures the error-disabled recovery timeout interval to 120 seconds.

```
device(config)# errdisable recovery interval 120
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | The **loam-critical-event** option was introduced. |
| 08.0.30mb | The **pvstplus-protect** option was introduced. |

# esn-enable (IPsec)

Used with replay-protection in IPsec to enable 64-bit sequence numbering for encrypted packets for tracking and verification by the receiving IPsec endpoint.

## Syntax

**esn-enable**

**no esn-enable**

## Command Default

Extended sequence numbering (ESN) is disabled by default.

## Modes

IPsec proposal configuration sub-mode

## Usage Guidelines

Configure ESN as part of the IPsec proposal.

ESN must be used in conjunction with replay-protection (configured in the IPsec profile).

Clear IPsec security associations (SAs) for the command to take effect.

The **no** form of the command disables ESN.

## Examples

The following example enables ESN in the IPsec proposal ipsecprop1.

```
device# configure terminal
device(config)# ipsec proposal ipsecprop1
device(config-ipsec-proposal-ipsecprop1)# esn-enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# ethernet (EFM-OAM)

Enables or disables EFM-OAM on an interface or multiple interfaces.

## Syntax

**ethernet** *unit/slot/port* [ [ **to** *unit/slot/port* ] [ **ethernet** *unit/slot/port* [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ] ] { **active** | **passive** | **allow-loopback** | **remote-failure critical-event action block-interface** }

**no ethernet** *unit/slot/port* [ [ **to** *unit/slot/port* ] [ **ethernet** *unit/slot/port* [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ] ] { **active** | **passive** | **allow-loopback** | **remote-failure critical-event action block-interface** }

## Command Default

The EFM-OAM is disabled locally on an interface.

## Parameters

**ethernet** *unit/slot/port*
Specifies the interface.

**to**
Configures the range of interfaces to enable EFM-OAM.

**lag** *lag-id*
Specifies the LAG virtual interface.

**active**
Sets the EFM-OAM operational mode as active on the interface.

**passive**
Sets the EFM-OAM operational mode as passive on the interface.

**allow-loopback**
Enables the interface to respond to a loopback request from the remote device.

**remote-failure critical-event action block-interface**
Configures the device to block the remote interface upon reception of a critical event information from the remote interface.

## Modes

EFM-OAM protocol configuration mode

## Usage Guidelines

When the active mode is specified, the device can send OAMPDU packets over the port to initiate an EFM-OAM discovery process. For the discovery process to be initiated, the EFM-OAM protocol must be enabled.

When the passive mode is specified, the device cannot use the port to send OAMPDU packets, but can respond if it receives OAMPDUs from the remote device.

When both peers are in passive mode (abnormal configuration), EFM-OAM protocol will not converge.

The OAMPDUs and pause frames will not be looped back in the loopback mode. All other Layer 2 protocol packets will be looped back if received on a loopbacked interface.

The **no** form of the command disables the EFM-OAM locally on the specified interface.

# Examples

The following example enables EFM-OAM on an interface and sets it to active mode.

```
device(config)# link-oam
device(config-link-oam)# ethernet 1/1/3 active
```

The following example enables EFM-OAM on a range of interfaces and sets them to active mode.

```
device(config)# link-oam
device(config-link-oam)# ethernet 1/1/4 to 1/1/8 active
```

The following example enables EFM-OAM on an interface and sets it to passive mode.

```
device(config)# link-oam
device(config-link-oam)# ethernet 2/1/1 passive
```

The following example enables EFM-OAM on a range of interfaces and sets them to passive mode.

```
device(config)# link-oam
device(config-link-oam)# ethernet 2/1/1 to 2/1/6 passive
```

The following example configures the interface to respond to the loopback request from the remote device.

```
device(config)# link-oam
device(config-link-oam)# ethernet 1/1/3 allow-loopback
```

The following example sets the device to block the interface when a critical event failure condition is detected.

```
device(config)# link-oam
device(config-link-oam)# ethernet 2/1/1 remote-failure critical-event action block-interface
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |
| 08.0.61 | This command was modified to add **lag** *lag-id* option. |

# ethernet loopback

Enables the Ethernet loopback functionality on a port in the VLAN-unaware mode.

## Syntax

**ethernet loopback**

**no ethernet loopback**

## Command Default

Ethernet loopback is not enabled on a port.

## Modes

Interface configuration mode

## Usage Guidelines

The Ethernet loopback functionality on a port in the VLAN-unaware mode can be configured either as flow-aware or flow-unaware. The specified port does not need to be explicitly assigned as a member of any VLAN.

To enable Ethernet loopback on a port in the VLAN-unaware mode as flow-aware, the **ethernet loopback test-mac** command must be executed before enabling the Ethernet loopback. The **ethernet loopback test-mac** command is mandatory on ICX 7750, ICX 7450, and ICX 7250 devices. To enable Ethernet loopback on these devices, you must first configure the **ethernet loopback test-mac** command. In other supported platforms, the **ethernet loopback test-mac** command is optional to enable Ethernet loopback.

To add or delete a port from VLAN, the VLAN unaware ethernet loopback configuration on the port must be removed.

Before adding or deleting a port form VLAN, the VLAN unaware ethernet configuration must be removed, if configured.

The **ethernet loopback** command is not supported on multiple ports (MIF) mode.

The **no** form of the command disables the Ethernet loopback functionality on the specified port.

## Examples

The following example configures Ethernet loopback on a specific port in the VLAN-unaware mode as flow-unaware.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ethernet loopback
```

The following example configures Ethernet loopback in VLAN-unaware mode as flow-aware.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ethernet loopback test-mac 1111.2222.3333 4444.5555.5555
device(config-if-e1000-1/1/1)# ethernet loopback
```

The following example shows the error which occurs when you try to add a port to VLAN, without removing the VLAN unaware ethernet loopback configuration.

```
bkes_oct14-16_DND(config-if-e1000-3/1/4)#vlan 10
bkes_oct14-16_DND(config-vlan-10)#tag eth 3/1/4
Error: Port 3/1/4 has Ethernet loopback configuration
Note: Remove Ethernet loopback from port 3/1/4 and then add port as member of VLAN 10
bkes_oct14-16_DND(config-vlan-10)#int eth 3/1/4
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# ethernet loopback (VLAN-aware)

Configures the Ethernet loopback functionality on one or a set of ports in a specific VLAN (VLAN-aware mode).

## Syntax

**ethernet loopback ethernet** *unit/slot/port* [ [ **to** *unit/slot/port* ] [ **ethernet** *unit/slot/port* ]... ]

**no ethernet loopback ethernet** *unit/slot/port* [ [ **to** *unit/slot/port* ] [ **ethernet** *unit/slot/port* ]... ]

**ethernet loopback lag** *lag-id* [ **to** *lag-id* | **ethernet** *unit/slot/port* [ [ **to** *unit/slot/port* ] [ **ethernet** *unit/slot/port* ]... ] ]

**no ethernet loopback lag** *lag-id* [ **to** *lag-id* | **ethernet** *unit/slot/port* [ [ **to** *unit/slot/port* ] [ **ethernet** *unit/slot/port* ]... ] ]

## Command Default

Ethernet loopback is not enabled on any port in a VLAN.

## Parameters

**ethernet**
Specifies the Ethernet interface.

**to**
Configures the range of ports.

*unit/slot/port*
Specifies the interface details.

**lag** *lag-id*
Specifies the LAG virtual interface.

## Modes

VLAN configuration mode

## Usage Guidelines

The Ethernet loopback functionality on a port in the VLAN-aware mode can be configured either as flow-aware or flow-unaware. The ports on which Ethernet loopback is being enabled must be explicitly assigned as a member of the VLAN.

To enable Ethernet loopback on a port in the VLAN-aware mode as flow-aware, the **ethernet loopback test-mac** command must be executed for the specific port from the interface mode before enabling Ethernet loopback. The **ethernet loopback test-mac** command is mandatory on ICX 7750, ICX 7450, and ICX 7250 devices. To enable Ethernet loopback on these devices, you must first configure the **ethernet loopback test-mac** command. In other supported platforms, the **ethernet loopback test-mac** command is optional to enable Ethernet loopback.

Enable **acl-per-port-per-vlan** configuration before issuing the ethernet loopback command. If not enabled, an error message "Error - Enable **acl-per-port-per-vlan** and configure VLAN unaware ethernet loopback" prompts you to enable the configuration.

A port cannot be configured as VLAN-aware and VLAN-unaware simultaneously, and the flow configuration must be either flow-aware or flow-unaware.

The **ethernet loopback** command in VLAN-aware mode is not supported on VLAN Group, VLAN Range, or mult-range VLAN (MVLAN) mode.

The **ethernet loopback** command VLAN-aware mode cannot be configured on a set of VLANs that share a Layer 2 topology (Topology Group).

The **no** form of the command disables Ethernet loopback from the ports of the specified VLAN.

# Examples

The following example configures Ethernet loopback in VLAN-aware mode as flow-aware.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ethernet loopback test-mac 1111.2222.3333 4444.5555.5555
device(config-if-e1000-1/1/1)# exit
device(config)# vlan 100
device(config-vlan-100)# ethernet loopback ethernet 1/1/1 to 1/1/10
```

The following example configures Ethernet loopback on a port in VLAN-aware mode as flow-unaware.

```
device(config)# vlan 100
device(config-vlan-100)# ethernet loopback ethernet 1/1/1
```

The following example configures Ethernet loopback on a range of ports in VLAN-aware mode as flow-unaware.

```
device(config)# vlan 100
device(config-vlan-100)# ethernet loopback ethernet 1/1/1 to 1/1/10
```

The following example configures Ethernet loopback on two separate ports in VLAN-aware mode as flow-unaware.

```
device(config)# vlan 100
device(config-vlan-100)# ethernet loopback ethernet 1/1/1 ethernet 1/2/3
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |
| 08.0.61 | This command was modified to add **lag** *lag-id* option. |

# ethernet loopback test-mac

Configures the port as flow-aware by specifying the source and destination MAC addresses of the flow on the interface.

## Syntax

**ethernet loopback test-mac** *destination-MAC source-MAC*

**no ethernet loopback test-mac** *destination-MAC source-MAC*

## Command Default

The port is flow-unaware.

## Parameters

*destination-MAC*
Specifies the flow parameter destination MAC address of the traffic.

*source-MAC*
Specifies the flow parameter source MAC address of the traffic.

## Modes

Interface configuration mode

## Usage Guidelines

You must configure the **ethernet loopback test-mac** command on ICX 7750, ICX 7450, and ICX 7250 devices before enabling Ethernet loopback. In other supported platforms, configure the **ethernet loopback test-mac** command only if you require the port to be flow-aware.

The source MAC address and destination MAC address must be unicast MAC addresses and the source MAC address must be unique across the network for proper Ethernet loopback operation.

You cannot configure a port as flow-aware and flow-unaware simultaneously. The flow can be configured on an in-service Ethernet loopback port. However, the flow configuration cannot be modified or removed if there is an ongoing loopback service on the interface.

The **ethernet loopback test-mac** command is not supported in multi-range VLAN (MVLAN) mode.

The **no** form of the command removes the flow configuration for the specified port.

## Examples

The following example configures the flow on a specific port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ethernet loopback test-mac 1111.2222.3333/4444.5555.5555
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# exclude ethernet

Excludes a port from the protocol VLAN membership.

## Syntax

**exclude ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ...]

**no exclude ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ...]

## Command Default

The port is not excluded from the protocol VLAN membership.

## Parameters

*stackid/slot/port*
Specifies the Ethernet port which should be excluded from the static protocol VLAN membership.

**to** *stackid/slot/port*
Specifies the range of ports that should be exluded from the static protocol VLAN membership.

## Modes

IP protocol VLAN configuration mode

IPX protocol VLAN configuration mode

IPv6 protocol VLAN configuration mode

AppleTalk protocol VLAN configuration mode

DECnet protocol VLAN configuration mode

NetBIOS protocol VLAN configuration mode

Other protocol VLAN configuration mode

## Usage Guidelines

The **no** form of the command includes in the protocol VLAN membership.

## Examples

The following example shows how to exclude ports from the protocol VLAN memebership.

```
device(config)# vlan 10
device(config-vlan-10)# atalk-proto name Red
device(config-atalk-proto)# no dynamic
device(config-atalk-proto)# exclude ethernet 1/1/1 to 1/1/3
```

# excluded-address

Specifies the addresses that should be excluded from the address pool.

## Syntax

**excluded-address** { *address* | *address-low address-high* }

## Parameters

*address*
Specifies a single address.

*address-low address-high*
Specifies a range of addresses.

## Modes

DHCP server pool configuration mode.

## Usage Guidelines

Use this command to specify either a single address or a range of addresses that are to be excluded from the address pool.

## Examples

The following example specifies the excluded address.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# excluded-address 10.2.3.44
```

# execute batch

Issues the commands that are saved in the batch buffer immediately or at a scheduled time, count, and interval.

## Syntax

**execute batch** *batch-id* { **after** *duration* | **at** *at-time at-date* | **cancel** }

**execute batch** *batch-id* **now** [ { **count** *count-value* | **end** *end-date* [ *end-time* ] } [ **interval** { **days** *value* | **hours** *value* | **mins** *value* } ] ]

**execute batch** *batch-id* **begin** *start-date* [ [ *start-time* ] [ **end** *end-date* [ *end-time* ] ] [ **interval** { **days** *value* | **hours** *value* | **mins** *value* } ] | **count** *count-value* [ **interval** { **days** *value* | **hours** *value* | **mins** *value* } ] | **end** *end-date* ]

## Command Default

Batch execution of CLI commands is not enabled.

## Parameters

*batch-id*

Specifies the unique batch buffer ID.

**after** *duration*

Schedules to run the commands in a batch after a specified period of time. The duration can be configured up to a maximum period of 49 days from the current system clock time. The duration is specified in the dd:hh:mm format indicating that the commands will be executed after the specified number of days, hours and minutes respectively.

**at** *at-time at-date*

Schedules to run the commands in a batch at a specific time and date.

**cancel**

Cancels the configured schedule to run the commands in a batch.

**now**

Issues the commands immediately.

**count** *count-value*

Specifies the number of times the commands in a batch must run. The range for the number of iterations is from 1 through 50.

**end** *end-date*

Specifies the date on which the batch command execution must stop. An end-date must be specified, followed by an end-time which is optional.

*end-time*

Specifies the time at which the batch command execution must stop. The default end-time is 23:59:59 of the specified end-date.

**interval**

Specifies the time interval at which the commands in a batch must run. The default value is 30 minutes. The time interval can be specified in days, hours, or minutes.

**days** *value*
Specifies the time interval in days at which the commands in a batch must run. The range is from 1 through 16 days.

**hours** *value*
Specifies the time interval in hours at which the commands in a batch must run. The range is from 1 through 24 hours.

**mins** *value*
Specifies the time interval in minutes at which the commands in a batch must run. The range is from 1 through 60 minutes.

**begin**
Schedules to run the commands in a batch from a specific date.

*start-date*
Specifies the date on which the batch command execution must start.

*start-time*
Specifies the time at which the batch command execution must start. The default start-time is 00:00:00 of the specified start-date.

# Modes

Privileged EXEC mode

# Usage Guidelines

At a particular instance, a batch can be scheduled only once.

A batch buffer cannot be scheduled when the batch execution process for that batch is in progress.

When a telnet or SSH session executing a batch command is closed, the corresponding batch execution will be cancelled.

Any command that requires user intervention will fail during batch execution.

The **no batch buffer** *batch-id* command from the global configuration mode removes the configured batch.

# Examples

The following example runs the commands that are saved in the batch buffer after 5 days, 3 hours, and 1 minute from the current system clock time.

```
device# execute batch 1 after 05:03:01
```

The following example runs the commands that are saved in the batch buffer at 04:05 AM on December 22, 2015.

```
device# execute batch 1 at 04:05:00 22-12-15
```

The following example runs the commands that are saved in the batch buffer immediately.

```
device# execute batch 1 now
```

The following example runs the commands that are saved in the batch buffer immediately and for a total of 5 times at an interval of 30 minutes (default interval).

```
device# execute batch 1 now count 5
```

The following example runs the commands that are saved in the batch buffer immediately and for a total of 5 times at an interval of 2 hours.

```
device# execute batch 1 now count 5 interval hours 2
```

The following example runs the commands that are saved in the batch buffer immediately and continues to execute the batch at an interval of 30 minutes (default interval) until 11:59 PM and 59 seconds (default end-time) on December 22, 2015.

```
device# execute batch 1 now end 12-22-15
```

The following example runs the commands that are saved in the batch buffer immediately and continues to execute the batch at an interval of 30 minutes (default interval) until 10:20 AM on December 22, 2015.

```
device# execute batch 1 now end 12-22-15 10:20:00
```

The following example runs the commands that are saved in the batch buffer immediately and continues to execute the batch at an interval of 4 days until 10:20 AM on December 22, 2015.

```
device# execute batch 1 now end 12-22-15 10:20:00 interval days 4
```

The following example cancels the configured schedule to issue the commands in a batch.

```
device# execute batch 1 cancel
```

The following example runs the commands that are saved in the batch buffer infinitely starting from 12 AM (midnight) (default start-time) on December 22, 2015 at an interval of 30 minutes (default interval).

```
device# execute batch 1 begin 12-22-15
```

The following example runs the commands that are saved in the batch buffer infinitely starting from 12 AM (midnight) (default start-time) on December 22, 2015 at an interval of 4 hours.

```
device# execute batch 1 begin 12-22-15 interval hours 4
```

The following example runs the commands that are saved in the batch buffer starting from 12 AM (midnight) (default start-time) on December 10, 2015 and continues to execute the batch at an interval of 30 minutes (default interval) until 11:59 PM and 59 seconds (default end-time) on December 22, 2015

```
device# execute batch 1 begin 12-10-15 end 12-22-15
```

The following example runs the commands that are saved in the batch buffer infinitely starting from 3:20 AM on December 22, 2015 at an interval of 30 minutes (default interval).

```
device# execute batch 1 begin 12-22-15 03:20:00
```

The following example runs the commands that are saved in the batch buffer infinitely starting from 3:20 AM on December 22, 2015 at an interval of 3 days.

```
device# execute batch 1 begin 12-22-15 03:20:00 interval days 3
```

The following example runs the commands that are saved in the batch buffer starting from 3:20 AM on December 10, 2015 and continues to execute the batch at an interval of 30 minutes (default interval) until 11:59 PM and 59 seconds (default end-time) on December 22, 2015.

```
device# execute batch 1 begin 12-10-15 03:20:00 end 12-22-15
```

The following example runs the commands that are saved in the batch buffer starting from 3:20 AM on December 10, 2015 and continues to execute the batch at an interval of 4 hours until 4:10 AM on December 22, 2015.

```
device# execute batch 1 begin 12-10-15 03:20:00 end 12-22-15 04:10:00 interval hours 4
```

The following example runs the commands that are saved in the batch buffer starting from 12 AM (midnight) (default start-time) on December 10, 2015 and for a total of 5 times at an interval of 30 minutes (default interval).

```
device# execute batch 1 begin 12-10-15 count 5
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | The **begin** keyword and corresponding options were introduced. Also, options such as **count** and **end** were added to the **now** keyword. |

# extend vlan add (VXLAN)

Configures a VLAN to be extended over the VXLAN tunnel to the designated remote site.

## Syntax

**extend vlan add** *vlan id*

**no extend vlan add** *vlan id*

## Command Default

VLAN extension is not set.

## Parameters

*vlan id*

Specifies the VLAN to be extended to the VXLAN remote site.

## Modes

Overlay-gateway site configuration mode

## Usage Guidelines

The **no** form of the command removes the extended VLAN configuration.

The command is supported only on ICX 7750 devices.

The VLAN must already be mapped to a VNI before it is extended over the gateway.

## Examples

The following example sets VLAN 10 to be extended over the remote site.

```
device# configure terminal
device(config)# overlay-gateway gate
device(config-overlay-gw-gate1)# map vlan 10 to VNI 888
device(config-overlay-gw-gate1)#site site1
device(config-overlay-gw-gate1-site1)# extend vlan add 10
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# external-lsdb-limit (OSPFv2)

Configures the maximum size of the external link state database (LSDB).

## Syntax

**external-lsdb-limit** *value*

**no external-lsdb-limit**

## Parameters

*value*

Maximum size of the external LSDB. Valid values range from 1 through 14913080. The default is 14913080.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

If you change the value, make sure to save the running-config file and reload the software. The change does not take effect until you reload or reboot the software.

The **no** form of the command restores the default setting.

## Examples

The following example sets the limit of the LSDB to 20000.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# external-lsdb-limit 20000
```

# external-lsdb-limit (OSPFv3)

Configures the maximum size of the external link state database (LSDB).

## Syntax

**external-lsdb-limit** *value*

**no external-lsdb-limit**

## Parameters

*value*

Maximum size of the external LSDB. Valid values range from 1 through 250000. The default is 250000.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

If you change the value, you must save the running-config file and reload the software. The change does not take effect until you reload or reboot the software.

The **no** form of command reverts to the default setting.

## Examples

The following example sets the limit of the external LSDB to 15000.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# external-lsdb-limit 15000
```

# failover

Enables or disables LAG (Link Aggregation Group) hardware failover on the next port in the LAG or on all ports in the LAG.

## Syntax

**failover** {**next** | **all**}

**no failover** {**next** | **all**}

## Command Default

LAG hardware failover is disabled.

## Parameters

**next**

Specifies that failover is to be enabled or disabled on the next port in the LAG.

**all**

Specifies that failover is to be enabled or disabled on all ports in the LAG.

## Modes

Dynamic LAG configuration mode

## Usage Guidelines

The **no** form of this command disables LAG hardware failover.

LAG hardware failover is supported only on Ruckus ICX 7750 devices.

## Examples

The following example enables LAG failover on the next port in the LAG:

```
device(config)# lag one dynamic
device(config-lag-one)# failover next
```

The following example enables LAG failover on all ports in the LAG:

```
device(config)# lag one dynamic
device(config-lag-one)# failover all
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# fast-external-fallover

Resets the session if a link to an eBGP peer goes down.

## Syntax

**fast-external-fallover**

**no fast-external-fallover**

## Modes

BGP configuration mode

## Usage Guidelines

Use this command to terminate and reset external BGP sessions of a directly adjacent peer if the link to the peer goes down, without waiting for the timer, set by the BGP **timers** command, to expire. This can improve BGP conversion time, but can also lead to instability in the BGP routing table as a result of a flapping interface.

## Examples

The following example configures the device to reset the session if a link to an eBGP peer goes down.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# fast-external-fallover
```

# fast port-span

Enables Fast Port Span, configuring the ports attached to the end stations to enter into the forwarding state in four seconds.

## Syntax

**fast port-span** [ **exclude** { **ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ] ]

**no fast port-span** [ **exclude ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ] ]

## Command Default

Fast Port Span is enabled by default on all ports that are attached to end stations.

## Parameters

**exclude**

Excludes a port from Fast Port Span while leaving Fast Port Span enabled globally.

**ethernet** *stackid/slot/port*

Specifies the Ethernet port that you want to exclude from Fast Port Span.

**to** *stackid/slot/port*

Specifies a range of Ethernet ports that you want to exclude from Fast Port Span.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

Disabling and then re-enabling Fast Port Span clears the exclude settings and thus enables Fast Port Span on all eligible ports. To make sure Fast Port Span remains enabled on the ports following a system reset, save the configuration changes to the startup-config file after you re-enable Fast Port Span. Otherwise, when the system resets, those ports will again be excluded from Fast Port Span.

Fast Port Span allows faster convergence on ports that are attached to end stations and thus do not present the potential to cause Layer 2 forwarding loops. Because the end stations cannot cause forwarding loops, they can safely go through the STP state changes (blocking to listening to learning to forwarding) more quickly than is allowed by the standard STP convergence time. Fast Port Span performs the convergence on these ports in four seconds (two seconds for listening and two seconds for learning).

Fast Port Span is a system-wide parameter and is enabled by default. Thus, when you boot a device, all the ports that are attached only to end stations run Fast Port Span. For ports that are not eligible for Fast Port Span, such as ports connected to other networking devices, the device automatically uses the normal STP settings.

The **no** form of the command disables Fast Port Span. Using the **exclude** option with the **no** form of the command enables Fast Port Span on the specified ports.

# Examples

The following example enables Fast Port Span on all ports.

```
device(config)# fast port-span
```

The following example excludes a set of ports from Fast Port Span.

```
device(config)# fast port-span exclude ethernet 1/1/1 ethernet 1/2/1 ethernet 1/3/1
```

The following example shows how to re-enable Fast Port Span on port 1/1/1 only while not re-enabling other excluded ports.

```
device(config)# no fast port-span exclude 1/1/1
```

The following example shows how to re-enable Fast Port Span on all excluded ports.

```
device(config)# no fast port-span
device(config)# fast port-span
device(config)# write memory
```

# fast uplink-span

Enables Fast Uplink Span, configuring a device deployed as a wiring closet switch to decrease the convergence time for the uplink ports to another device to just one second.

## Syntax

**fast uplink-span ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ...]

**no fast uplink-span ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ...]

## Command Default

Fast Uplink Span is not enabled.

## Parameters

**ethernet** *stackid/slot/port*

Specifies the Ethernet port on which you want to enable Fast Uplink Span.

**to** *stackid/slot/port*

Specifies a range of ports on which you want to enable Fast Uplink Span.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

The new uplink port goes directly to forward mode (bypassing listening and learning modes). The wiring closet switch must be a Ruckus device, but the device at the other end of the link can be a Ruckus device or another vendor's switch.

To configure Fast Uplink Span, specify a group of ports that have redundant uplinks on the wiring closet switch (Ruckus device). If the active link becomes unavailable, Fast Uplink Span transitions the forwarding to one of the other redundant uplink ports in just one second. All Fast Uplink Span-enabled ports are members of a single Fast Uplink Span group.

To avoid the potential for temporary bridging loops, it is recommended that you use Fast Uplink Span only for wiring closet switches (switches at the edge of the network cloud). In addition, enable Fast Uplink Span only on a group of ports intended for redundancy, so that at any given time only one of the ports is expected to be in the forwarding state.

The **no** form of the command removes Fast Uplink Span on the ports.

## Examples

The following example configures a group of ports for Fast Uplink Span.

```
device(config)# fast uplink-span ethernet 1/4/1 to 1/4/4
```

The following example configures Fast Uplink Span for a VLAN.

```
device(config)# vlan 10
device(config-vlan-10)# untag ethernet 1/8/1 to 1/8/2
device(config-vlan-10)# fast uplink-span ethernet 1/8/1 to 1/8/2
```

# fdp advertise

Configures the IP management address to advertise for Foundry Discovery Protocol (FDP) neighbors.

## Syntax

**fdp advertise** { **ipv4** | **ipv6** }

**no fdp advertise** { **ipv4** | **ipv6** }

## Command Default

When FDP is enabled, by default, the device advertises one IPv4 address and one IPv6 address to its FDP neighbors.

## Parameters

**ipv4**

Advertises only the IPv4 management address.

**ipv6**

Advertises only the IPv6 management address.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command sets the device to advertis one IPv4 address and one IPv6 address to its FDP neighbors.

When FDP is enabled, by default, the device advertises one IPv4 address and one IPv6 address to its FDP neighbors. If desired, you can configure the device to advertise only the IPv4 management address or only the IPv6 management address. You can set the configuration globally on a Layer 2 switch or at the interface level on a Layer 3 switch.

## Examples

The following example configures the device to advertise only the IPv6 management address.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# fdp advertise ipv6
```

# fdp enable

Enables Foundry Discovery Protocol (FDP) on an interface.

## Syntax

**fdp enable**

**no fdp enable**

## Command Default

FDP is enabled at the interface level once FDP is enabled on the device.

## Modes

Interface configuration mode

## Usage Guidelines

When FDP is enabled globally, you can disable and re-enable FDP on individual ports.

The **no** form of the command disables FDP on an interface.

## Examples

The following example enables FDP on an interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# fdp enable
```

# fdp holdtime

Configures the Foundry Discovery Protocol (FDP) update hold time.

## Syntax

**fdp holdtime** *secs*

**no fdp holdtime**

## Command Default

By default, a device that receives an FDP update holds the information until the device receives a new update or until 180 seconds have passed since receipt of the last update.

## Parameters

*secs*

Specifies the number of seconds for which a device that receives an FDP update can hold the update before discarding it. Valid values are from 10 through 255. The default value is 180.

## Modes

Global configuration mode

## Usage Guidelines

Once the device receives a new update or once 180 seconds have passed since receipt of the last update, the device discards the update.

The **no** form of the command sets the hold time to its default value of 180 seconds.

## Examples

The following example sets the FDP hold time to 200 seconds.

```
device(config)# fdp holdtime 200
```

# fdp run

Enables a device to send Foundry Discovery Protocol (FDP) packets globally.

## Syntax

**fdp run**

**no fdp run**

## Command Default

FDP is disabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables the sending of FDP packets.

## Examples

The following example enables FDP globally.

```
device(config)# fdp run
```

# fdp timer

Configures the Foundry Discovery Protocol (FDP) update timer.

## Syntax

**fdp timer** *secs*

**no fdp timer**

## Command Default

By default, a device enabled for FDP sends an FDP update every 60 seconds.

## Parameters

*secs*

Specifies the number of seconds between FDP updates. The value can range from 5 through 900 seconds. The default value is 60 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the FDP timer to its default value of 60 seconds.

## Examples

The following example sets the FDP timer to 360 seconds.

```
device(config)# fdp timer 360
```

# fingerprint (PKI)

Sets the authentication fingerprint for the Certificate Authority (CA).

## Syntax

**fingerprint** { *fingerprint_value* }

**no fingerprint**

## Command Default

## Parameters

*fingerprint_value*

ASCII string in the format xx:xx:xx:x... that defines the CA fingerprint.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

## Examples

The following configuration for the trustpoint trust1 configures the authentication fingerprint for the CA as the value shown.

```
device# configure terminal
device(config)# pki trustpoint trust1
device(config-pki-trustpoint-trust1)# auto-enroll
device(config-pki-trustpoint-trust1)# enrollment retry-period 2
device(config-pki-trustpoint-trust1)# enrollment profile profile1
device(config-pki-trustpoint-trust1)# pki-entity entity1
device(config-pki-trustpoint-trust1)# eckeypair key-label eckeyAuto
device(config-pki-trustpoint-trust1)# fingerprint 36:0c:92:6e:df:b2:72:eb:59:e8:63:73:2a:98:a8:91:cb:
50:94:d9
device(config-pki-trustpoint-trust1)# ocsp http post
device(config-pki-trustpoint-trust1)# exit
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# filter-strict-security enable

Enables or disables strict filter security for MAC authentication and 802.1X authentication.

## Syntax

**filter-strict-security**

**no filter-strict-security**

## Command Default

Strict filter security is enabled.

## Modes

Authentication mode

## Usage Guidelines

When strict security mode is enabled, authentication for a port fails if the Filter-Id attribute contains invalid information, or if insufficient system resources are available to implement the IP ACLs.

When strict security mode is enabled:

- If the Filter-Id attribute in the Access-Accept message contains a value that does not refer to an existing filter (that is, IP ACL configured on the device), then the client will not be authorized, regardless of any other information in the message (for example, if the Tunnel-Private-Group-ID attribute specifies a VLAN on which to assign the port).
- If the device does not have the system resources available to dynamically apply a filter to a port, then the client will not be authenticated.

When strict filter security is disabled:

- If the Filter-Id attribute in the Access-Accept message contains a value that does not refer to an existing filter (that is, a MAC address filter or IP ACL configured on the device), then the client remains authorized and no filter is dynamically applied to it.
- By default, strict security mode is enabled for all MAC authentication and 802.1X-enabled interfaces, but you can manually disable or enable it using the **filter-strict-security** command from the authentication configuration mode or using the **authentication filter-strict-security** command from the interface configuration mode.

The **no** form of the command disables strict filter security.

## Examples

The following example enables strict filter security.

```
device(config)# authentication
device(config-authen)# filter-strict-security enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30mb | This command was modified. |

# flash

Use the **flash** command to perform basic flash file maintenance.

## Syntax

**flash** { **copy** *source-file destination-file* | **dbgflock** | **delete** *flash-file* | **files** *directory-name* | **rename** *source-file destination-file* }

## Parameters

**copy** *source-file destination-file*
Copy the source flash file to a new file

**dbgflock**
Display the flash access lock holder

**delete** *flash-file*
Delete the flash file

**files** *directory-name*
Display flash files in a particular directory

**rename** *source-file destination-file*
Rename a flash file

## Modes

Privileged EXEC mode

## Usage Guidelines

The command is useful in flash file maintenance.

## Examples

In the following example, flash files are displayed.

```
device# flash files
Type       Size   Name
----------------------
F        24108665 primary
F        24108665 secondary
F             610 startup-config.backup
F            2052 startup-config.txt

48219992 bytes 4 File(s) in FI root

     1768706048 bytes free in FI root
     1768706048 bytes free in /
```

The **show flash** command also displays flash file information but with different results.

```
device# show flash
Stack unit 1:
  Compressed Pri Code size = 24108665, Version:08.0.40qT213 (SPR08040q074.bin)
  Compressed Sec Code size = 24108665, Version:08.0.40qT213 (SPR08040q074.bin)
  Compressed Boot-Monitor Image size = 786944, Version:10.1.05T215
  Code Flash Free Space = 1768706048
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# flash-timeout

Configures the flash timeout duration.

## Syntax

**flash-timeout** *time*

**no flash-timeout** *time*

## Command Default

The default flash timeout value is 12 minutes.

## Parameters

*time*

Specifies the flash timeout value in minutes and the range is from 12 to 60 minutes.

## Modes

Global configuration mode

## Usage Guidelines

The new timeout value will be effective from the next flash operation.

The **no** form of the command removes the flash timeout configuration and restores the default value of 12 minutes.

## Examples

The following example configures the flash timeout value as 30 minutes.

```
device(config)# flash-timeout 30
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# flow-control

Enables or disables flow control and flow control negotiation, and advertises flow control.

## Syntax

**flow-control** [ **both** | **generate-only** | **honor-only** ]

**no flow-control** [ **both** | **generate-only** | **honor-only** ]

## Command Default

Flow control is enabled.

## Parameters

**both**

Flow control in PAUSE generation and honoring mode.

**generate-only**

Flow control in PAUSE generation only mode.

**honor-only**

Flow control in PAUSE honoring (Default) mode.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

On ICX 7750 devices the default packet-forwarding method is cut-through, in which port flow control (IEEE 802.3x) is not supported but priority-based flow control (PFC) is supported. You can configure the **store-and- forward** command in global configuration mode to enable the store-and-forward method for packet-forwarding.

The recommended flow control settings when the ICX 7750 switch is set for store-and-forward, are listed below.

| Symmetrical flow control | Port-based flow control | Configuration commands | Cut-through (Jumbo enabled) | **Store-and-Forward (Jumbo Enabled)** |
|---|---|---|---|---|
| disabled | honor-only | # **flow-control honor-only** | Not Recommended | OK |
| | no flow-control-both | #**no flow-control both** | OK | OK |
| enabled | no flow-control-both | #**symmetrical-flow-control enable**<br># **no flow-control both** | OK | OK |
| | honor-only | # **symmetrical-flow-control enable**<br>#**flow-control honor-only** | Not Recommended | OK |
| | generate-only | # **symmetrical-flow-control enable** | OK | OK |

| Symmetrical flow control | Port-based flow control | Configuration commands | Cut-through (Jumbo enabled) | **Store-and-Forward (Jumbo Enabled)** |
|---|---|---|---|---|
| | | # **flow-control generate-only** | | |
| | both | # **symmetrical-flow-control enable**<br>#**flow-control both** | Not Recommended | OK |

By default, when flow control is enabled globally and auto-negotiation is on, flow control is enabled and advertised on 10/100/1000M ports. If auto-negotiation is off or if the port speed was configured manually, flow control is neither negotiated with nor advertised to the peer.

**NOTE**
Enabling only port auto-negotiation does not enable flow control negotiation. You must use the **flow-control neg-on** command to enable flow-control negotiation.

The **no** form of the command disables flow control.

# Examples

The following example disables flow control globally.

```
device(config)# no flow-control
```

The following example enables flow control on ethernet ports 1/1/11 to 1/1/15.

```
device(config)# interface ethernet 1/1/11 to 1/1/15
device(config-mif-1/1/11-1/1/15)# flow-control
```

The following example disables flow control on ethernet port 1/1/9.

```
device(config)# interface ethernet 1/1/9
device(config-if-e1000-1/1/9)# no flow-control
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was modified. Enabling only auto-negotiation does not enable flow-control negotiation. |

# force-up ethernet

Forces the member port of a dynamic LAG (Link Aggregation Group) to be logically operational even if the dynamic LAG is not operating.

## Syntax

**force-up ethernet** *port*

**no force-up ethernet** *port*

## Command Default

The member ports of a dynamic LAG are logically operational only if the dynamic LAG is operating.

## Parameters

*port*

Specifies the port.

## Modes

Dynamic LAG configuration mode

## Usage Guidelines

The **no** form of the command causes the specified port to be logically operational only when the dynamic LAG is operating.

When the dynamic LAG is not operational, the port goes to "force-up" mode. In this mode, the port is logically operational, which enables a PXE-capable host to boot from the network using this port. Once the host successfully boots from the network, the dynamic LAG can connect the host to the network with the LAG link. Even if the dynamic LAG fails later, this port is brought back to "force-up" mode and remains logically operational.

A port that is in "force-up" mode has the operational status ("Ope" ) of "Frc". Use the **show lag** command to display the operational status.

If any port in a dynamic LAG receives an LACPDU, the port in force-up mode leaves force-mode and becomes a member port in the dynamic LAG.

## Examples

The following example enables PXE boot support on member port 3/1/1 of a dynamic LAG R4-dyn.

```
device(config)# lag R4-dyn
device(config-lag-R4-dyn)# force-up ethernet 3/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced. |

# format disk0

Formats the external USB.

## Syntax

**format disk0**

## Modes

User EXEC mode.

## Examples

The following example formats the external USB.

```
device# format disk0
Are you sure?(enter 'y' or 'n'): formatting The External USB (disk0) of size 64.2GB
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# gig-default

Configures the Gbps fiber negotiation mode on individual ports, overriding the global configuration mode.

## Syntax

**gig-default** { **neg-full-auto** | **auto-gig** | **neg-off** }

**no gig-default** { **neg-full-auto** | **auto-gig** | **neg-off** }

## Command Default

The globally configured Gbps negotiation mode is the default mode for all Gbps fiber ports.

## Parameters

**neg-full-auto**

Configures the port to first try to perform a handshake with the other port to exchange capability information. If the other port does not respond to the handshake attempt, the port uses the manual configuration (or the defaults if an administrator has not configured the information). That is, the device performs autonegotiation first and if it is failed then performs non-autonegotiation. This is the default.

**auto-gig**

Configures the port to try to perform a handshake with the other port to exchange capability information.

**neg-off**

Configures the port to not try to perform a handshake. Instead, the port uses information that was manually configured by an administrator.

## Modes

Interface configuration mode

## Usage Guidelines

**NOTE**

When Gbps negotiation mode is turned off (CLI command **gig-default neg-off**), the device may inadvertently take down both ends of a link. This is a hardware limitation for which there is currently no workaround.

The **no** form of the command resets the configuration to the default of trying to perform a handshake with other ports to exchange capability information.

## Examples

The following example sets the negotiation mode to auto-gig mode for ports 1/1/1 to 1/1/4.

```
device(config)# interface ethernet 1/1/1 to 1/1/4
device(config-mif-1/1/1-1/1/4)# gig-default auto-gig
```

# graceful-restart (BGP)

Enables the BGP graceful restart capability.

## Syntax

**graceful-restart** [ **purge-time** *seconds* | **restart-time** *seconds* | **stale-routes-time** *seconds* ]

**no graceful-restart** [ **purge-time** *seconds* | **restart-time** *seconds* | **stale-routes-time** *seconds* ]

## Command Default

Graceful restart is enabled globally.

## Parameters

**purge-time** *seconds*

Specifies the maximum period of time, in seconds, for which a restarting device maintains stale routes in the BGP routing table before purging them. Range is from 1 to 3600 seconds. The default value through 600 seconds.

**restart-time** *seconds*

Specifies the restart time, in seconds, advertised to graceful-restart-capable neighbors. Range is from 1 through 3600 seconds. The default value is 120 seconds.

**stale-routes-time** *seconds*

Specifies the maximum period of time, in seconds, that a helper device will wait for an End-of-RIB (EOR) marker from a peer. All stale paths are deleted when this time period expires. Range is from 1 through 3600 seconds. The default value is 360 seconds.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

Use this command to enable or disable the graceful restart capability globally for all BGP neighbors in a BGP network. If the graceful restart capability is re-enabled after a BGP session has been established, the neighbor session must be cleared for GR to take effect.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

Use this command in BGP configuration mode to disable or re-enable the BGP4 graceful restart capability globally, or to alter the default parameters. Use this command in address-family IPv6 unicast configuration mode to disable or re-enable the BGP4+ graceful restart capability globally or to alter the default parameters.

The **purge-time** parameter is applicable for both restarting and helper devices. The timer starts when a BGP connection is closed. The timer ends when an EOR is received from all nodes, downloaded into BGP and an EOR sent to all neighbors. The configured purge-time timer value is effective only on the configured node.

The **restart-time** parameter is applicable only for helper devices. The timer starts at the time the BGP connection is closed by the remote peer and ends when the Peer connection is established. The configured restart time timer value is effective only on the peer node, and not in the configured node. During negotiation time, the timer value is exchanged.

The **stale-routes-time** parameter is applicable only for helper devices. The timer starts when the peer connection is established once the HA-failover peer node has been established. The timer ends at the time an EOR is received from the peer. The configured stale-time timer value is effective only on the configured node.

Use the **clear ip bgp neighbor** command with the **all** parameter for the changes to the GR parameters to take effect immediately.

The **no** form of the command disables the BGP graceful restart capability globally for all BGP neighbors.

# Examples

The following example disables the BGP4 graceful restart capability.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# no graceful-restart
```

The following example re-enables the BGP4 graceful restart capability.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1.1.1.1 remote-as 2
device(config-bgp-router)# graceful-restart
```

The following example disables the BGP4+ graceful restart capability.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# no graceful-restart
```

The following example re-enables the BGP4+ graceful restart capability.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1000::1 remote-as 2
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 1000::1 activate
device(config-bgp-ipv6u)# graceful-restart
```

The following example sets the purge time to 240 seconds at the IPv4 address family configuration level.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1.1.1.1 remote-as 2
device(config-bgp-router)# graceful-restart
%Warning: Please clear the neighbor session for the parameter change to take effect!
device(config-bgp-router)# graceful-restart purge-time 240
```

The following example sets the restart time to 60 seconds at the IPv4 address family configuration level.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1.1.1.1 remote-as 2
device(config-bgp-router)# graceful-restart
%Warning: Please clear the neighbor session for the parameter change to take effect!
device(config-bgp-router)# graceful-restart restart-time 60
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

The following example sets the stale-routes time to 180 seconds at the IPv6 address family configuration level.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 1
device(config-bgp-router)# neighbor 1000::1 remote-as 2
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 1000::1 activate
device(config-bgp-ipv6u)# graceful-restart
%Warning: Please clear the neighbor session for the parameter change to take effect!
device(config-bgp-ipv6u)# graceful-restart stale-routes-time 180
%Warning: Please clear the neighbor session for the parameter change to take effect!
```

# graceful-restart (OSPFv2)

Enables the OSPF Graceful Restart (GR) capability.

## Syntax

**graceful-restart** [ **helper-disable** | **restart-time** *seconds* ]

**no graceful-restart**

## Command Default

Graceful restart and graceful restart helper capabilities are enabled.

## Parameters

**helper-disable**
Disables the GR helper capability.

**restart-time**
Specifies the maximum restart wait time, in seconds, advertised to neighbors. The default value is 120 seconds. The configurable range of values is from 10 through 1800 seconds.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use **no graceful-restart helper-disable** to re-enable the GR helper capability.

The **no** form of the command disables the graceful restart capability.

## Examples

The following example disables the GR helper capability.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# graceful-restart helper-disable
```

The following example re-enables the GR helper capability.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# no graceful-restart helper-disable
```

The following example re-enables the GR capability.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# graceful-restart
```

The following example re-enables the GR capability and changes the maximum restart wait time from the default value to 240 seconds.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# graceful-restart restart-time 240
```

# graceful-restart helper (OSPFv3)

Enables the OSPFv3 graceful restart (GR) helper capability.

## Syntax

**graceful-restart helper** { **disable** | **strict-lsa-checking** }

**no graceful-restart helper**

## Command Default

GR helper is enabled.

## Parameters

**disable**

Disables the OSPFv3 GR helper capability.

**strict-lsa-checking**

Enables the OSPFv3 GR helper mode with strict link-state advertisement (LSA) checking.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command disables the GR helper capability on a device.

## Examples

The following example enables GR helper and sets strict LSA checking.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router-ospf)# graceful-restart helper strict-lsa-checking
```

# graft-retransmit-timer

Configures the time between the transmission of graft messages sent by a device to cancel a prune state.

## Syntax

**graft-retransmit-timer** *seconds*

**no graft-retransmit-timer** *seconds*

## Command Default

The graft retransmission time is 180 seconds.

## Parameters

*seconds*

Specifies the time in seconds. The range is 60 through 3600 seconds. The default is 180 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the default graft retransmission time, 180 seconds.

Messages sent by a device to cancel a prune state are called graft messages. When it receives a graft message, the device responds with a Graft Ack (acknowledge) message. If this Graft Ack message is lost, the device that sent it resends it.

## Examples

This example configures a graft retransmission timer to 90 seconds.

```
device(config)# router pim
device(config-pim-router)# graft-retransmit-timer 90
```

# group-router-interface

Creates router interfaces for each VLAN in the VLAN group.

## Syntax

**group-router-interface**

**no group-router-interface**

## Command Default

A group router interface is not configured.

## Modes

VLAN group configuration mode

## Usage Guidelines

The **group-router-interface** command creates router interfaces for each VLAN in the VLAN group by using the VLAN IDs of each of the VLANs as the corresponding virtual interface number. This command enables a VLAN group to use a virtual routing interface group. You can enter this command when you configure the VLAN group for the first time or later, after you have added tagged ports to the VLAN, and so on.

If a VLAN group contains VLAN IDs greater than the maximum virtual interface number allowed, the **group-router-interface** command will be rejected.

The **no** form of the command disables the VLAN group router interface.

## Examples

The following example shows how to create a router interface for a VLAN.

```
device(config)# vlan-group 1 vlan 10
device(config-vlan-group-1)# group-router-interface
```

# gvrp-base-vlan-id

Configures a VLAN ID as a base VLAN for GVRP.

## Syntax

**gvrp-base-vlan-id** *vlan-id*

**no gvrp-base-vlan-id** *vlan-id*

## Command Default

GVRP uses VLAN 4093 as the base VLAN for the protocol.

## Parameters

*vlan-id*

Configures the new base VLAN. You can specify a VLAN ID from 2 through 4092 or 4095.

## Modes

Global configuration mode

## Usage Guidelines

All ports that are enabled for GVRP become tagged members of the base VLAN. If you need to use VLAN 4093 for a statically configured VLAN, you can change the GVRP base VLAN ID.

**NOTE**
If you want to change the GVRP base VLAN ID, you must do so before enabling GVRP.

The **no** form of the command changes the base VLAN to the default VLAN ID of 4093.

## Examples

The following example shows how to configure a new base VLAN for GVRP.

```
device(config)# gvrp-base-vlan-id 1001
```

# gvrp-enable

Enables the Generic Attribute Registration Protocol (GARP) VLAN Registration Protocol (GVRP) and enters GVRP configuration mode.

## Syntax

**gvrp-enable**

**no gvrp-enable**

## Command Default

GVRP is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

Single STP must be enabled on the device. Ruckus implementation of GVRP requires Single STP. If you do not have any statically configured VLANs on the device, you can enable Single STP.

The **no** form of the command disables GVRP.

## Examples

The following example shows how to enable Single STP and then GVRP.

```
device(config)# vlan 1
device(config-vlan-1)# spanning-tree
device(config-vlan-1)# exit
device(config)# spanning-tree single
device(config)# gvrp-enable
device(config-gvrp)#
```

# gvrp-max-leaveall-timer

Configures the maximum value for the minimum interval at which GVRP sends Leaveall messages on all GVRP interfaces.

## Syntax

**gvrp-max-leaveall-timer** *time*

**no gvrp-max-leaveall-timer** *time*

## Command Default

The default value is 300,000 milliseconds (ms).

## Parameters

*time*

Specifies the maximum time in milliseconds to which you want to set the Leaveall timer. You can specify from 300,000 to 1,000,000 (one million) milliseconds. The value must be a multiple of 100 ms.

## Modes

Global configuration mode

## Usage Guidelines

Enter this command before enabling GVRP. Once GVRP is enabled, you cannot change the maximum Leaveall timer value. By default, you can set the Leaveall timer to a value five times the Leave timer - the maximum value allowed by the software (configurable from 300000 ms to 1000000 ms).

This command does not change the default value of the Leaveall timer itself. The command only changes the maximum value to which you can set the Leaveall timer.

The **no** form of the command changes the maximum time value to the default value.

## Examples

The following example shows how to set the maximum value for the Leaveall timer.

```
device(config)# gvrp-max-leaveall-timer 1000000
```

# hardware-drop-disable

Disables passive multicast route insertion (PMRI).

## Syntax

**hardware-drop-disable**

**no hardware-drop-disable**

## Command Default

PMRI is enabled.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the default and enables PMRI.

To prevent unwanted multicast traffic from being sent to the CPU, PIM routing and PMRI can be used together to ensure that multicast streams are forwarded out only on ports with interested receivers and unwanted traffic is dropped in hardware on Layer 3 switches. To disable this process, use the **hardware-drop-disable** command.

**NOTE**
Disabling hardware-drop does not immediately take away existing hardware-drop entries, they will go through the normal route aging processing when the traffic stops.

## Examples

This example disables PMRI.

```
device(config)#router pim
device(config-pim-router)# hardware-drop-disable
```

# hello-interval (VRRP)

Configures the interval at which master Virtual Router Redundancy Protocol (VRRP) routers advertise their existence to the backup VRRP routers.

## Syntax

**hello-interval** [ **msec** ] *interval*

**no hello-interval** [ **msec** ] *interval*

## Command Default

Hello messages from VRRP master routers are sent to backup routers every second.

## Parameters

**msec** *interval*

Interval, in milliseconds, at which a master VRRP router advertises its existence to the backup VRRP routers. Valid values range from 100 through 84000. The default is 1000. VRRP-E does not support the hello message interval in milliseconds.

*interval*

Sets the interval, in seconds, for which a VRRP backup router waits for a hello message from the VRRP master router before determining that the master is offline. Valid values range from 1 through 84. The default value is 1.

## Modes

VRID interface configuration mode

## Usage Guidelines

A VRRP master router periodically sends hello messages to the backup routers. The backup routers use the hello messages as verification that the master is still online. If the backup routers stop receiving the hello messages for the period of time specified by the dead interval, the backup routers determine that the master router is dead. At that point, the backup router with the highest priority becomes the new master router.

By default, the dead interval is internally derived from the hello interval. It is equal to 3 times the hello interval plus the skew time, where the skew time is equal to (256 minus the priority) divided by 256. Generally, if you change the hello interval on the master VRRP router using the **hello-interval** command, you also should also change the dead interval on the VRRP backup routers using the **dead-interval** command.

The **hello-interval** command is configured only on master VRRP routers and is supported by VRRP and VRRP-E.

The **no** form resets the hello message interval to its default value of 1000 milliseconds (1 second).

**NOTE**
VRRP-E does not support the hello message interval in milliseconds.

# Examples

The following example enables advertisements from the VRRP master router and sets the hello message interval to 10,000 milliseconds.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner
device(config-if-e1000-1/1/6-vrid-1)# ip address 10.53.5.1
device(config-if-e1000-1/1/6-vrid-1)# hello-interval msec 10000
device(config-if-e1000-1/1/6-vrid-1)# activate
VRRP router 1 for this interface is activating
```

The following example enables advertisements from the VRRP-E master router and sets the hello message interval to 15 seconds.

```
device# configure terminal
device(config)# router vrrp-extended
device(config)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.5.3/24
device(config-if-e1000-1/1/5)# ip vrrp-extended vrid 2
device(config-if-e1000-1/1/5-vrid-2)# backup priority 50 track-priority 10
device(config-if-e1000-1/1/5-vrid-2)# ip-address 10.53.5.1
device(config-if-e1000-1/1/5-vrid-2)# hello-interval 15
device(config-if-e1000-1/1/5-vrid-2)# activate
VRRP router 2 for this interface is activating
```

# hello-interval (VSRP)

Configures the number of seconds between hello messages from the master to the backups for a given VRID.

## Syntax

**hello-interval** { **msec** *interval* | *interval* }

**no hello-interval** { **msec** *interval* | *interval* }

## Command Default

Hello messages from master are sent to backup every second.

## Parameters

**msec** *interval*

Interval, in milliseconds, at which a master advertises its existence to the backup. Valid values range from 100 through 40900. The default is 1000.

*interval*

Sets the interval, in seconds, for which a backup waits for a hello message from the master before determining that the master is offline. Valid values range from 1 through 84. The default value is 1.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The Master periodically sends hello messages to the backup. The backup routers use the hello messages as verification that the master is still online. If the backup routers stop receiving the hello messages for the period of time specified by the dead interval, the backup routers determine that the master router is dead. At that point, the backup router with the highest priority becomes the new master router.

By default, the dead interval is internally derived from the hello interval. It is equal to 3 times the hello interval plus one-half second. Generally, if you change the hello interval on the master router using the **hello-interval** command, you also should also change the dead interval on the backup routers using the **dead-interval** command.

The **no** form resets the hello message interval to its default value of 1000 milliseconds (1 second).

# Examples

The following example sets the hello message interval to 10,000 milliseconds.

```
device# configure terminal
device(config)# vlan 400
device(config-vlan-400)# tagged ethernet 1/1/4 to 1/1/9
device(config-vlan-400)# vsrp vrid 4
device(config-vlan-400-vrid-4)# hello-interval msec 10000
```

The following example sets the hello message interval to 15 seconds.

```
device# configure terminal
device(config)# vlan 400
device(config-vlan-400)# tagged ethernet 1/1/4 to 1/1/9
device(config-vlan-400)# vsrp vrid 4
device(config-vlan-400-vrid-4)# hello-interval 15
```

# hello-timer

Configures the interval at which hello messages are sent out of Protocol Independent Multicast (PIM) interfaces.

## Syntax

**hello-timer** *seconds*

**no hello-timer** *seconds*

## Command Default

The hello interval is 30 seconds.

## Parameters

*seconds*

Specifies the interval in seconds. The range is 10 through 3600 seconds. The default is 30 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the default hello interval, 30 seconds.

Devices use hello messages to inform neighboring devices of their presence.

## Examples

This example configures a hello interval of 120 seconds on all ports on a device operating with PIM.

```
device(config)# router pim
device(config-pim-router)# hello-timer 120
```

# hitless-failover enable

Enables hitless stacking failover and switchover. The standby controller is allowed to take over the active role without reloading the stack when failover occurs.

## Syntax

**hitless-failover enable**

**no hitless-failover enable**

## Command Default

Hitless stacking failover is enabled. In earlier releases, failover and switchover were disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no** form of the command to disable hitless stacking failover. The change takes effect immediately.

The **hitless-failover enable** and **no hitless-failover enable** commands must be executed from the active stack controller.

You must assign a stack mac address to the device using the **stack mac address** command before you can execute the **hitless-failover enable** command.

## Examples

The following example enables hitless stacking switchover and failover on the active controller for the stack.

```
device(config)# hitless-failover enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |
| 08.0.20 | Hitless failover is enabled by default. |

# hold-down-interval

Configures the hold-down interval.

## Syntax

**hold-down-interval** *number*

**no hold-down-interval** *number*

## Command Default

The default hold-down time interval is 3 seconds.

## Parameters

*number*

The time interval for the new master to hold the traffic. The time interval ranges from 1 through 84 seconds.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The hold-down interval prevents the occurrence of Layer 2 loops during failover by delaying the new master from forwarding traffic long enough to ensure that the failed master is unavailable.

The **no** form of the command sets the time interval to the default value.

## Examples

The following example shows how to change the hold-down interval.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# hold-down-interval 4
```

# host-max-num

Limits the number of hosts that are authenticated at any one time.

## Syntax

**host-max-num** *number*

**no host-max-num** *number*

## Command Default

There is no limit to the number of hosts that can be authenticated (0).

## Parameters

*number*

Specifies the number of hosts that can be authenticated at any one time. The valid values are from 0 through 8192. The default is 0, that is there is no limit to the number of hosts that can be authenticated.

## Modes

Web Authentication configuration mode

## Usage Guidelines

The maximum number of hosts that can be authenticated at one time is 8192 or the maximum number of MAC addresses the device supports. When the maximum number of hosts has been reached, the device redirects any new host that has been authenticated successfully to the Maximum Host web page.

The **no** form of the command sets no limit (default).

## Examples

The following example limits the number of hosts that can be authenticated at one time to 10.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# host-max-num 10
```

# hostname

Configures a system name for a device and saves the information locally in the configuration file for future reference.

## Syntax

**hostname** *string*

**no hostname**

## Command Default

The device has a factory-set hostname.

## Parameters

*string*

Configures the system name. The name can be up to 255 alphanumeric characters. The host name should be enclosed in quotation marks if it contains spaces.

## Modes

Global configuration mode

## Usage Guidelines

When you configure a system name, the name replaces the default system name in the CLI command prompt.

The **no** form of the command removes the configured hostname.

## Examples

The following example configures a system name.

```
device(config)# hostname headquarters
headquarters(config)#
```

# Commands I

## ignore-temp-shutdown

Prevents shutdown of ICX 7150, ICX 7450, and ICX 7750 devices at the threshold shutdown temperature.

### Syntax

**ignore-temp-shutdown**

**no ignore-temp-shutdown**

### Command Default

By default, the function is disabled.

### Modes

Global configuration mode

Stack-unit configuration mode

### Usage Guidelines

The command is applicable only on ICX 7150, ICX 7450, and ICX 7750 devices.

Use the **no** form of the command to re-enable shutdown based on temperature threshold. The **no** form of the command disables the battleshort mode at global level and at unit level.

Either the global battleshort mode or unit-specific battleshort mode is enabled but not both.

This command can be executed at a global level and at a unit level. If the command is enabled or disabled at global level, it applies to all the units which are part of the stack. If the command is enabled or disabled at a unit level, it applies only to that unit alone in the stack. To execute this command at a unit level, specify the unit ID at the configuration mode.

### Examples

The following example enables battleshort mode on a standalone device or globally on all stack units.

```
device(config)# ignore-temp-shutdown
Ignore temperature shutdown threshold has been enabled
```

The following example enables battleshort mode on an individual stack member.

```
device# configure terminal
device(config)# stack unit 2
device(config-unit-2)# ignore-temp-shutdown
Ignore temperature shutdown threshold has been enabled in Stack unit 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.60 | The command was introduced. |
| 08.0.61 | The command was updated to support members of an ICX 7150 stack. |

# ike-profile

Configures an IKEv2 profile for an IPsec profile.

## Syntax

**ike-profile** *name*

**no ike-profile** *name*

## Command Default

The default IKEv2 profile is **def-ike-profile**.

## Parameters

*name*

Specifies the name of an IKEv2 profile.

## Modes

IPsec profile configuration mode

## Usage Guidelines

When an IPsec profile is created, it is automatically configured to use the default IKEv2 profile. Use this command to configure an alternate IKEv2 profile for the IPsec profile.

The **no** form of the command restores the default IKEv2 profile configuration for the IPsec profile.

## Examples

The following example shows how to configure an IKEv2 profile named ikev2_prof for an IPsec profile named ipsec_prof.

```
device(config)# ipsec profile ipsec_prof
device(config-ipsec-profile-ipsec_prof)# ike-profile ikev2_prof
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 auth-proposal

Creates an Internet Key Exchange version 2 (IKEv2) authentication proposal and enters configuration mode for the proposal.

## Syntax

**ikev2 auth-proposal** *auth-name*

**no ikev2 auth-proposal** *auth-name*

## Parameters

*auth-name*

Specifies the name of an IKEv2 authentication proposal.

## Modes

Global configuration mode

## Usage Guidelines

An IKEv2 authentication proposal defines the authentication methods used in IKEv2 peer negotiations.

An IKEv2 authentication proposal is activated by attaching it to an IKEv2 profile.

The **no** form of the command removes the IKEv2 authentication proposal configuration.

## Examples

The following example shows how to create an IKEv2 authentication proposal named "secure" and enters configuration mode for the proposal.

```
device# configure terminal
device(config)# ikev2 auth-proposal secure
device(config-ike-auth-proposal-secure)#
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 exchange-max-time

Configures the maximum setup time for Internet Key Exchange version 2 (IKEv2) message exchange.

## Syntax

**ikev2 exchange-max-time** *seconds*

**no ikev2 exchange-max-time** *seconds*

## Command Default

The default value is 30 seconds.

## Parameters

*seconds*

Specifies the maximum setup time in seconds. The time range is from 1 through 300 seconds.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command resets the maximum setup time to the default value.

## Examples

The following example sets the maximum setup time for IKEv2 message exchange to 50 seconds.

```
device# configure terminal
device(config)# ikev2 exchange-max-time 50
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 limit

Configures limits for the number of Internet Key Exchange version 2 (IKEv2) security association (SA) sessions.

## Syntax

**ikev2 limit** { **max-in-negotiation-sa** *limit* | **max-sa limit** *limit* }

**no ikev2 limit** { **max-in-negotiation-sa** *limit* | **max-sa limit** *limit* }

## Command Default

The default limit (for each type of SA session) is 256.

## Parameters

**max-in-negotiation-sa** *limit*

Limits the total number of in-negotiation IKEv2 SA sessions. The range is from 1 through 256.

**max-sa** *limit*

Limits the total number of IKEv2 SA sessions. The range is from 1 through 256.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command returns the specified SA session limit to the default value.

## Examples

The following example shows how to limit the maximum number of in-negotiation IKEv2 SA sessions to 10.

```
device# configure terminal
device(config)# ikev2 limit max-in-negotiation-sa 10
```

The following example shows how to limit the maximum number of IKEv2 SA sessions to 200.

```
device# configure terminal
device(config)# ikev2 limit max-sa 200
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 policy

Creates an Internet Key Exchange version 2 (IKEv2) policy and enters IKEv2 policy configuration mode.

## Syntax

**ikev2 policy** *name*

**no ikev2 policy** *name*

## Command Default

The default IKEv2 policy is **def-ike-policy**.

## Parameters

*name*

Specifies the name of an IKEv2 policy.

## Modes

Global configuration mode

## Usage Guidelines

There is a default IKEv2 policy (**def-ike-policy**) that is used to protect IKEv2 SA negotiations. The default policy does not require configuration and has the following settings:

- **proposal**: **def-ike-prop**
- **local_address**: Not set; matches all local addresses
- **vrf**: Not set; matches the default-VRF

Use the **ikev2 policy** command to configure any additional IKEv2 policies that you need.

The **no** form of the command removes any IKEv2 policy configuration other than the default IKEv2 policy.

The default IKEv2 policy cannot be removed.

Only one IKEv2 policy can be selected for a local endpoint (single IPv4 address). Configuring multiple IKEv2 policies for the same IP address is invalid.

When multiple matching policies are identified during IKEv2 negotiations, the most recently created matching policy is used.

## Examples

The following example creates an IKEv2 policy named test_policy1.

```
device# configure terminal
device(config)# ikev2 policy test_policy1
device(config-ike-policy-test_policy1)#
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 profile

Creates an Internet Key Exchange version 2 (IKEv2) profile and enters IKEv2 profile configuration mode.

## Syntax

**ikev2 profile** *name*

**no ikev2 profile** *name*

## Command Default

The default IKEv2 profile is **def-ike-profile**.

## Parameters

*name*

Specifies the name of an IKEv2 profile.

## Modes

Global configuration mode

## Usage Guidelines

An IKEv2 profile defines the local and peer identities and the authentication proposal for an IKEv2 session.

The default IKEv2 profile (**def-ike-profile**) does not require configuration and has the following settings:

- **authentication**: **def-ike-auth-prop**
- **protected**: Any
- **local-identifier address**: 0.0.0.0
- **lifetime**: 2592000
- **keepalive**: 300

Use the **ikev2 profile** command to configure any additional IKEv2 profiles.

The **no** form of the command removes any IKEv2 profile configuration other than the default IKEv2 profile.

The default IKEv2 profile cannot be removed.

## Examples

The following example shows how to create an IKEv2 profile named ikev2_profile1.

```
device# configure terminal
device(config)# ikev2 profile ikev2_profile1
device(config-ike-profile-ikev2_profile1)#
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 proposal

Creates an Internet Key Exchange version 2 (IKEv2) proposal and enters IKEv2 proposal configuration mode.

## Syntax

**ikev2 proposal** *name*

**no ikev2 proposal** *name*

## Command Default

The default IKEv2 proposal is **def-ike-proposal**.

## Parameters

*name*

Specifies the name of an IKEv2 proposal.

## Modes

Global configuration mode

## Usage Guidelines

An IKEv2 proposal defines a set of algorithms that are used in IKEv2 peer negotiations.

There is a default IKEv2 proposal (**def-ike-proposal**) that does not require configuration and has the following settings:

- **encryption**: AES-CBC-256
- **prf**: SHA-384
- **integrity**: SHA-384
- **dh-group**: 20

Use the **ikev2 proposal** command to configure any additional IKEv2 proposals.

The default IKEv2 proposal configuration cannot be removed.

## Examples

The following example shows how to create an IKEv2 proposal named test_proposal1.

```
device# configure terminal
device(config)# ikev2 proposal test_proposal1
device(config-ike-proposal-test_proposal1)#
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 retransmit-interval

Configures the delay time for resending Internet Key Exchange version 2 (IKEv2) messages.

## Syntax

**ikev2 retransmit-interval** *time*

**no ikev2 retransmit-interval** *time*

## Command Default

The default delay time is 5 seconds.

## Parameters

*time*

Specifies the delay time in seconds. The time ranges from 1 through 60.

## Modes

Global configuration mode

## Usage Guidelines

The retransmit interval increases exponentially.

The **no** form of the command restores the default value.

## Examples

The following example show how to configure the delay time for resending IKEv2 messages to 20 seconds.

```
device# configure terminal
device(config)# ikev2 retransmit-interval 20
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ikev2 retry-count

Configures the maximum number of attempts to retransmit an Internet Key Exchange version 2 (IKEv2) message.

## Syntax

**ikev2 retry-count** *number*

**no ikev2 retry-count** *number*

## Command Default

The default number of attempts is 5.

## Parameters

*number*

Specifies the maximum number of attempts to retransmit an IKE message. The range is from 1 through 25.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the retry count to the default value.

## Examples

The following example shows how to configure the number of retry attempts for transmitting an IKEv2 message to 8.

```
device# configure terminal
device(config)# ikev2 retry-count 8
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# image-auto-copy disable

Turns off the auto image copy function used in a stack or an 802.1br (bridge port extension) configuration to restore all units to the same software image.

## Syntax

**image-auto-copy disable**

**no image-auto-copy disable**

## Command Default

Auto image copy is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

Use this command when you want to have manual control over image updates in the stack or bridge port extension domain.

The auto image copy process is not triggered if the major versions of the mismatched units are not the same. For example, if the image version is 8.0.30 in the mismatched unit, it cannot be automatically updated to 8.0.40. However, if an 8.0.40 image is present in the mismatched unit, and it needs to be updated to 8.0.40b, the auto image copy process works.

Use the **show stack detail** or the **show running-config** command to determine whether auto image copy is enabled.

The **no** form of the command re-enables auto image copy, which restarts immediately and ensures all stack units have the same image.

## Examples

The following example disables auto image copy.

```
device(config)# image-auto-copy disable
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# import-users

Imports a text file of user records from a TFTP server to the device.

## Syntax

**import-users tftp** *ip-address* **filename** *name*

## Parameters

**tftp** *ip-address*

Specifies the IP address of the TFTP server from which the file must be imported.

**filename** *name*

Specifies the name of the file to import from the TFTP server.

## Modes

Local user database configuration mode

## Usage Guidelines

Before importing the file, make sure it adheres to the ASCII text format. The text file to be imported must be in the following ASCII format.

```
[delete-all]
[no] username
username1
password
password1
cr
[no] username
username2
password
password2
cr
...
```

The **delete-all** command entry in the text file indicates that the user records in the text file will replace the user records in the specified local user database on the switch. If the **delete-all** entry is not present, the new user records will be added to the specified local user database on the switch. The **delete-all** command entry is optional. If present, it must appear on the first line, before the first user record in the text file. If you want to delete a user entry from the specified local user database on the switch, use the **no username** command entry in the text file. User records that already exist in the local user database will be updated with the information in the text file when it is uploaded to the switch. For username1, username2, and so on, enter up to 31 ASCII characters.

## Examples

The following example imports a text file of user records from a TFTP server.

```
device(config)# local-userdb userdb1
device(config-localuserdb-userdb1)# import-users tftp 192.168.1.1 filename userdb1
```

# inactivity-timer

Configures the time a forwarding entry can remain unused before the device deletes it.

## Syntax

**inactivity-timer** *seconds*

**no inactivity-timer** *seconds*

## Command Default

The default inactive time is 180 seconds.

## Parameters

*seconds*

Specifies the time in seconds. The range is 60 through 3600 seconds. The default is 180 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the default inactive time, 180 seconds.

A device deletes a forwarding entry if the entry is not used to send multicast packets. The Protocol Independent Multicast (PIM) inactivity timer defines how long a forwarding entry can remain unused before the device deletes it.

**NOTE**
The inactivity timer may not expire according to the configured time. You may notice a delay of 0 to 60 seconds over the configured value.

## Examples

This example configures an inactive time to 90 seconds.

```
device# configure terminal
device(config)# router pim
device(config-pim-router)# inactivity-timer 90
```

## History

| Release version | Command history |
|---|---|
| 08.0.60 | Added note about the inactivity timer expiry. |

# include-port

Adds ports to the VSRP.

## Syntax

**include-port ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ]

**no include-port ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ]

## Command Default

By default, all the ports on which you configure a VRID are interfaces for the VRID.

## Parameters

**ethernet** *stackid/slot/port*

Adds the Ethernet interface to the VRID.

**to** *stackid/slot/port*

Adds a range of Ethernet interfaces to the VRID.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

Removing a port is useful because there is no risk of a loop occurring, such as when the port is attached directly to an end host and you plan to use a port in a metro ring.

When a port is removed from VSRP, the port remains in the VLAN but its forwarding state is not controlled by VSRP.

The **no** form of the command removes the ports from VSRP.

## Examples

The following example shows how to remove a port from the VRID.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# no include-port ethernet 1/1/2
```

# initial-contact-payload

Configures sending an initial contact message to a peer for an Internet Key Exchange version 2 (IKEv2) profile.

## Syntax

**initial-contact-payload**

**no initial-contact-payload**

## Command Default

No initial contact message is sent to a peer for an IKEv2 profile.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

The initial contact message is sent to ensure that old security associations (SAs) on the peer are deleted. When a device reboots, peers may have security associations (SAs) that are no longer valid. The initial contact message ensures that any old SAs on the peer are deleted.

The **no** form of the command disables initial contact messages from being sent to a peer for an IKEv2 profile.

## Examples

The following example enables sending an initial contact message to a peer for an IKEv2 profile named ikev2_profile1.

```
device# configure terminal
device(config)# ikev2 profile ikev2_profile1
device(config-ike-profile-ikev2_profile1)# initial-contact-payload
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# initial-ttl

Configures the Hello packet time to live (TTL) (the number of hops a Hello message can traverse after leaving the device and before the Hello message is dropped).

## Syntax

**initial-ttl** *number*

**no initial-ttl** *number*

## Command Default

The default TTL is 2.

## Parameters

*number*

Specifies the number of hops a Hello message can traverse after leaving the device and before the Hello message is dropped. The range is from 1 through 255. The default value is 2.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

When a VSRP device (master or backup) sends a VSRP Hello packet, the device subtracts one from the TTL. Thus, if the TTL is 2, the device that originates the Hello packet sends it out with a TTL of 1. Each subsequent device that receives the packet also subtracts one from the packet TTL. When the packet has a TTL of 1, the receiving device subtracts 1 and then drops the packet because the TTL is zero.

A metro ring counts as one hop, regardless of the number of nodes in the ring.

The **no** form of the command sets the TTL to the default value.

## Examples

The following examples sets the TTL to 5.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# initial-ttl 5
```

# inline power

Configures inline power on Power over Ethernet (PoE) ports in interface configuration mode and link aggregation group (LAG) secondary ports in global configuration mode.

## Syntax

**inline power ethernet** *interface* [ **power-by-class** | *power-class* | **power-limit** *power-limit* | **priority** *priority -value* ]

**inline power ethernet** *interface* [ **power-by-class** | *power-class* | **power-limit** *power-limit* | **priority** *priority -value* ]

**NOTE**

The **ethernet***interface* pair of parameters is required only if you want to configure inline power on secondary ports (you must use global configuration mode to do this).

## Command Default

PoE is enabled by default and power is automatically allocated to all PoE-capable ports on bootup.

## Parameters

**ethernet**

Specifies an ethernet interface. You can configure the **ethernet** keyword only in global configuration mode.

*interface*

Specifies the number of the ethernet interface. This is used only with the **ethernet** keyword.

**power-by-class**

Specifies the power limit based on class value. The range is 0-4. The default is 0.

**power-limit**

Specifies the power limit based on actual power value in mW. The range is 1000-15400|30000mW. The default is 15400|30000mW. For PoH ports, the range is 1000-95000mW, and the default is 95000mW. The power-limit value is rounded to the nearest multiple of 5 on PoH ports.

**priority**

Specifies the priority for power management. The range is 1 (highest) to 3 (lowest). The default is 3.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

As the 'inline power' configuration is applied on all PoE-capable ports by default, PD is powered up as soon as it is connected to the port. If the PoE power allocation needs to be disabled on bootup, use the **no inline power** command and do write memory. Upon reboot, all the saved PoE configurations would get applied and PoE will not be enabled.

Data link operation is decoupled with inline power by default and this behavior cannot be altered through user configuration.

You cannot configure inline power on PoE LAG ports in interface configuration mode because the interface-level configuration is not available in the CLI for LAG secondary ports. The **inline power ethernet** command enables you to configure inline power on secondary ports in global configuration mode.

The **no** form of the command disables PoE.

# Examples

The following example configures inline power on LAG ports.

```
Device(config)# lag "mylag" static id 5
Device(config-lag-mylag)# ports ethernet 1/1/1 to 1/1/4
Device(config)#inline power ethernet 1/1/1 power-by-class 3
Device(config)#inline power ethernet 1/1/3 priority 2
Device(config)#inline power ethernet 1/1/4 power-limit 12000
```

# History

| Release | Command History |
|---|---|
| 08.0.01 | This command was modified to run in global configuration mode using the **ethernet** keyword. The **decouple-datalink** keyword was also introduced. |
| 08.0.20 | This command was modified to allow requisite PoH power limits. |
| 08.0.70 | This command was modified to change the default behavior to keep the PoE enabled. It was also modified to remove **decouple-datalink** keyword. |

# inline power adjust class

Use these commands when powered devices (PDs) are entering an overload state as a result of faulty PDs power requests.

## Syntax

**inline power adjust classr** *class* { **delta** *milliwatts* | **minimum** *milliwatts* }

**no inline power adjust classr** *class* { **delta** *milliwatts* | **minimum** *milliwatts* }

## Parameters

*class*
The detected PD class for which this configuration is applied to. Values ramge from 0 through 4.

**delta**
Tthe amount of extra power allocated above the LLDP/CDP requested power.

*milliwatts*
The additional allocated power measured in milliwatts.

**minimum**
The minimum power that must be allocated, even if the PD LLDP/CDP requested power is lower than the configuration.

## Modes

Global configuration mode

## Usage Guidelines

These configurations should be used only when ports are entering an overload condition because of faulty PDs that are requesting lower power through LLDP/CDP messages and then consuming higher than the requested power.

The delta option assures the power allocation is equal to LLDP/CDP requested power plus delta power that is configured for that PD class.

The minimum option assures that the power allocation is equal to the maximum of LLDP/CDP power requested and the minimum power configured for that PD class.

Given a configuration of **inline power adjust class** 1 **delta** 800. If a class 1 PD is connected and is requesting power of 2600 milliWatts through LLDP/CDP, then the total allocation from the switch would be 3200 milliWatts. But if a class 2 PD is connected then there won't be any extra power allocation. If you want the extra power allocation for a class 2 PD, the configuration would be **inline power adjust class** 2 **delta** 800.

## Examples

Set the detected PD class to 1 and allocate 800 milliwatts of extra power for the class.

```
device(config)# inline power adjust class 1 delta  800
```

Set the detected PD class to 1 and allocate minimum power (in milliwatts) regardless of the LLDP/CDP requested power level.

```
device(config)# inline power adjust class 1 minimum  3200
```

# History

| Release | Command History |
|---|---|
| 8.0.30f | This command was introduced. |

# inline power couple-datalink

Links the behavior of PoE configuration with interface disable or interface enable configuration.

## Syntax

**inline power couple-datalink**

**no inline power**

## Command Default

Data link operation is decoupled with inline power by default.

## Modes

Interface configuration mode

## Usage Guidelines

When they are linked, the **interface disable** command also removes the power on the port (disables power when interface is disabled).

The following are some datalink operations that can affect the operational state of the PoE on PoE ports when datalink coupling is enabled:

- Using the **disable** command on the power sourcing equipment (PSE) port interface.
- LAG operational changes can affect the PoE power state if datalink coupling is enabled. That is power on LAG ports are impacted when LAG is undeployed, when the **disable** command is issued on LAG port, or when an interface is deleted from the LAG.

In situations where datalink operations tamper with PoE configurations and disable the power on the port, the interface has to be enabled so as to get the power enabled.

To reinstate the default setting of datalink decouple configuration, user must configure the **inline power** command on the interface.

The **no** function of the **inline power couple-datalink** command does not restore the default setting but only disables the power on the port.

## Examples

The following example couples datalink operations with inline power.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# inline power couple-datalink
```

The following example reinstates the default datalink decouple configuration on an interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# inline power
```

# History

| Release | Command History |
|---|---|
| 08.0.70b | This command was introduced. |
| 08.0.80 | This command was added to FastIron 08.0.80 release. |

# inline power install-firmware

Installs Power over Ethernet (PoE) firmware.

## Syntax

**inline power install-firmware** { **all** | **spx-unit** *unit-number* | **stack-unit** *unit-number* } **tftp** *ip-address file-name*

## Parameters

**all**
Installs Firmware on all PoE units of the system.

**spx-unit** *unit-number*
Specifies the unit ID of the SPX unit.

**stack-unit** *unit-number*
Specifies the unit ID of the stack. If the switch is not a part of the stack, the unit number is the default value. The default stack-unit value is 1.

**tftp** *ip-address*
Specifies the IP address of the TFTP server.

*file-name*
Specifies the name of the file, including its path name.

## Modes

Privileged EXEC mode

## Usage Guidelines

In releases prior to 08.0.61, PoE firmware installation could be initiated only on one SPX unit or stack unit at a time.

From 08.0.61 release onwards, PoE firmware installation can also be initiated on all PoE units, or on multiple stack or SPX units simultaneously.

## Examples

The following example installs PoE firmware on all PoE units.

```
device# inline power install-firmware all tftp 10.120.54.161 icx74xx_poh_01.2.1.b003.fw
```

The following example installs PoE firmware on a stack unit.

```
device# inline power install-firmware stack-unit 1 tftp 10.120.54.161 icx74xx_poh_01.2.1.b003.fw
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | The command was modified to add the **all** keyword. |

# inline power install-firmware scp

Upgrades the PoE firmware of a FastIron stacking device by downloading a firmware file from an SCP server.

## Syntax

**inline power install-firmware** { **all** | **spx-unit** *unit-number* | **stack-unit** *unit-id* } **scp** { *ipv4-address-* | *ipv4-hostname-* | **ipv6** { *ipv6-address-* | *ipv6-hostname-* } **outgoing-interface** { **ethernet** *stackid/slot/port* | **ve** *ve-number* } } [ **public-key** { **dsa** | **rsa** } ] [ *remote-port* ] *remote-filename*

## Parameters

**all**
Installs Firmware on all PoE units of the system.

**spx-unit** *unit-number*
Specifies the unit ID of the SPX unit.

**stack-unit** *unit-id*
Specifies the unit ID of the FastIron device in the stack to copy the PoE firmware. You must specify the stack unit when you configure the **inline power install-firmware** command to upgrade PoE firmware on a stacking device.

**module** *module-id*
Specifies the module ID of the device to copy the PoE firmware.

*ipv4-address-*
Specifies the IPV4 address of the SCP server, using 8-bit values in dotted decimal notation.

*ipv4-hostname-*
Specifies the IP hostname of the SCP server.

**ipv6**
Specifies the IPV6 address method for SCP file transfer.

*ipv6-address-prefix/prefix-length*
Specifies the IPV6 address of the SCP server. You must specify this address in hexadecimal using 16-bit values between colons, as documented in RFC 2373.

*ipv6-hostname-*
Specifies the IPv6 hostname of the SCP server.

**outgoing-interface**
Specifies the interface to be used to reach the remote host.

**ethernet** *stackid/slot/port*
Configures an Ethernet interface as the outgoing interface.

**ve** *ve-number*
Configures a virtual interface (VE) as the outgoing interface.

**public-key**
Specifies the type of public key authentication to use for the connection, either digital signature algorithm (DSA) or Rivest, Shamir, and Adelman (RSA) . If you do not configure this parameter, the default authentication type is password.

**dsa**
Specifies DSA as the public key authentication.

**rsa**
Specifies RSA as the public key authentication.

*remote-port*
Specifies the remote port number for the TCP connection.

*remote-filename*
Specifies the name of the file in the SCP server that is be transferred. You can specify up to 127 characters for the filename.

# Modes

Privileged EXEC mode

# Usage Guidelines

You are prompted for username and password when you configure this command.

If you do not configure the type of public key authentication, the default authentication type is password.

In releases prior to 08.0.61, PoE firmware installation could be initiated only on one SPX unit or stack unit at a time.

From 08.0.61 release onwards, PoE firmware installation can also be initiated on all PoE units, or on multiple stack or SPX units simultaneously.

You must specify the stack unit and module when you configure the **inline power install-firmware** command to upgrade PoE firmware on a stacking device.

# Examples

This example upgrades the PoE firmware of a FastIron device by downloading a firmware file from an SCP server:

```
device# inline power install-firmware stack-unit 2 scp 2.2.2.2 icx64xx_poeplus_02.1.0.b004.fw
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.61 | The command was modified to add the **all** keyword. |

# inline power interface-mode-2pair-pse

Corrects a condition where some non-standard powered devices (PD) are undetected on PoH ports due to difference in allowed capacitance between a 2-pair port and a 4-pair port.

## Syntax

**inline power interface-mode-2pair-pse**

**no inline power interface-mode-2pair-pse**

## Modes

Interface port

## Usage Guidelines

This command is applicable for 4pair pse ports of all ICX platforms.

The 4pair ports are moved to AT mode when overdrive is disabled using the **no inline power overdrive** command and port is made 2pair.

Before this command is executed the user may see the following behavior:

```
SPX(config)# interface ethernet 17/1/8
SPX(config-if-pe-e1000-17/1/8)# enable
SYSLOG: <14> Sep 27 17:32:49 SPX PORT: 17/1/8 enabled by un-authenticated user
from console session.
SYSLOG: <14> Sep 27 17:32:49 SPX System: PoE: Allocated power of 95000 mwatts
on port 17/1/8.
SYSLOG: <14> Sep 27 17:32:56 SPX System: PoE: Released complete power of 95000
mwatts on port 17/1/8.
SPX(config-if-pe-e1000-17/1/8)# show inline power 17/1/8

 Port   Admin   Oper    ---Power(mWatts)---  PD Type  PD Class  Pri  Fault/
        State   State   Consumed  Allocated                        Error
-----------------------------------------------------------------------------
 17/1/8 On      Off            0          0  n/a      n/a         3  n/a
```

Where the Operating State is listed as Off, there is no power consumed or allocated, and the PD is not recognized..

# Examples

To correct this problem:

```
SPX# configure terminal
SPX(config)# interface ethernet 17/1/8
SPX(config-if-pe-e1000-17/1/8)# inline power interface-mode-2pair-pse
SYSLOG: <14> Sep 27 17:34:52 SPX System: PoE: Allocated power of 7000 mwatts on
port 17/1/8. PoE: Power enabled on port 17/1/8.
SYSLOG: <14> Sep 27 17:34:52 SPX System: PoE: Power enabled on port 17/1/8.
SYSLOG: <14> Sep 27 17:34:52 SPX System: Interface ethernet 17/1/8, state up
SPX(config-if-pe-e1000-17/1/8)# show running-config interface ethernet 17/1/8
interface ethernet 17/1/8
 spanning-tree root-protect
 spanning-tree 802-1w admin-edge-port
 inline power
 inline power interface-mode-2pair-pse
 stp-bpdu-guard
 trust dscp
 port security
 enable
!
SPX(config-if-pe-e1000-17/1/8)# show inline power 17/1/8

 Port   Admin   Oper    ---Power(mWatts)---  PD Type  PD Class  Pri  Fault/
        State   State   Consumed  Allocated                       Error
-----------------------------------------------------------------------
 17/1/8 On      On          2700       7000  Legacy   Class 2     3  n/a
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# inline power non-pd-detection enable

Enables detection for nonpowered endpoints or devices (non-PD).

## Syntax

**inline power non-pd-detection enable**

**no inline power non-pd-detection enable**

## Command Default

By default, non-PD detection is enabled.

## Modes

Global configuration mode.

## Usage Guidelines

A multiport PD must be connected to a single unit and must have a LAG defined for the ports.

By default, this feature is enabled and new devices that connect to the Power over Ethernet (PoE) ports are detected.

When this feature is disabled by using the **no** form of the command, new devices that connect to the PoE ports are not detected.

When this feature is re-enabled after having been disabled, only new devices that connect to the PoE ports are detected. To ensure that all existing non-PDs are detected, you must save the configuration and reload the device or follow the below order of configuration:

1. Configure the LAG for multiport PDs.
2. Enable non-PD detection mode.
3. Configure inline power on interfaces.

When the **no** form of the command is used to disable non-PD detection, the existing non-PD state declarations on the ports are not cleared. The state declarations on the ports clear when they are disconnected from the non-PDs or when you save the configuration and reload the device.

Either reload after disabling the mode or disable and then enable inline power on ports that are in a non-PD state.

When a port has detected a non-PD, it generates the following syslog message:

```
PoE: Power disabled on port 1/1/21 because of detection of non-PD.
PD detection will be disabled on port.
```

When a port loses a non-PD (cable disconnected, etc.), it generates the following syslog message:

```
PoE: Port 1/1/21 lost non-PD, so enabling PD detection.
```

# Examples

The following example disables non-PD detection.

```
device# configure terminal
device(config)# no inline power non-pd-detection enable
```

The following example enables non-PD detection.

```
device# configure terminal
device(config)# inline power non-pd-detection enable
Warning: Enabling or disabling non-PD detection requires reload or
disable/enable of ports with existing non-PDs.
Warning: Enabling this configuration also has following limitation:
All ports of a multi-port PD must be connected to one unit only so
that a LAG configured does not span more than a single unit.
device(config)# write memory
device(config)# exit
device# reload
```

# History

| Release version | Command history |
|---|---|
| 08.0.30f | This command was introduced. |
| 08.0.50 | The command was modified; non-PD detection is now enabled by default. |

# inline power overdrive

Allows the Class 0 and Class 4 PDs to negotiate for power greater than 30-watt allocation through LLDP protocol messages.

## Syntax

**inline power overdrive**

**no inline power overdrive**

## Command Default

PoE overdrive is enabled.

## Modes

Global configuration mode

## Usage Guidelines

The maximum power that can be processed based on LLDP negotiation is limited to the hardware capability of the PSE.

If the PD negotiates for power more than the hardware limit, the PSE allocates only up to the hardware capability of the PSE.

PoE overdrive is supported only on PoH and PoE+ ports.

Overdrive is valid only on 2-pair ports and 2-pair operation mode on 4-pair ports.

The **no** form of the command prevents the PDs from sending further power overdrive request. However, the power allocated to the PDs based on the earlier PoE overdrive request remains valid.

## Examples

The following example configures PoE overdrive.

```
device(config)# inline power overdrive
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# integrity

Configures an integrity algorithm for an Internet Key Exchange version 2 (IKEv2) proposal.

## Syntax

**integrity** { **sha256** | **sha384** }

**no integrity** { **sha256** | **sha384** }

## Command Default

The default integrity algorithm is SHA-384.

## Parameters

**sha256**

Specifies SHA-2 family 256-bit (hash message authentication code (HMAC) variant) as the hash algorithm.

**sha384**

Specifies SHA-2 family 384-bit (HMAC variant) as the hash algorithm.

## Modes

IKEv2 proposal configuration mode

## Usage Guidelines

Multiple integrity algorithms may be configured for an IKEv2 proposal.

When only one integrity algorithm is configured for an IKEv2 proposal, removing it restores the default configuration.

The **no** form of the command removes the specified integrity algorithm configuration.

## Examples

The following example shows how to configure the integrity algorithm SHA-256 for an IKEv2 proposal name ikev2_proposal.

```
device(config)# ikev2 proposal ikev2_proposal
device(config-ikev2-proposal-ikev2_proposal)# integrity sha256
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# interface ethernet

Enters interface configuration mode for the specified Ethernet interface.

## Syntax

**interface ethernet** *stackid/slot/port* [ [**ethernet** *stackid/slot/port* ]... | **to** *stackid/slot/port* ]

**no interface ethernet** *stackid/slot/port* [ [**ethernet** *stackid/slot/port.* ]... | **to** *stackid/slot/port* ]

## Parameters

**ethernet** *stackid/slot/port*

Specifies the Ethernet interface.

**to** *stackid/slot/port*

Specifies a range of Ethernet interfaces.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of the command exits from the interface configuration mode.

## Examples

The following example shows how to enter interface configuration mode.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# interface ethernet 1/1/2
```

The following example shows how to move to one interface mode to another.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# interface ethernet 1/1/2
device(config-if-e1000-1/1/2)#
```

# interface group-ve

Associates the virtual interface routing group with a VLAN group.

## Syntax

**interface group-ve** *num*

**no interface group-ve** *num*

## Command Default

A virtual routing interface group is not associated with a VLAN group.

## Parameters

*num*

Specifies the VLAN group ID with which you want to associate the virtual routing interface group.

## Modes

Global configuration mode

## Usage Guidelines

The VLAN group must already be configured and enabled to use a virtual routing interface group. The software automatically associates the virtual routing interface group with the VLAN group that has the same ID. You can associate a virtual routing interface group only with the VLAN group that has the same ID.

When you configure a virtual routing interface group, all members of the group have the same IP subnet address.

**NOTE**
Configuring a virtual interface routing group is not supported with IPv6. Configuring a virtual interface routing group is supported only with the OSPF, VRRPv2, and VRRP-Ev2 protocols.

The **no** form of the command removes the virtual routing interface group from a VLAN group.

## Examples

The following example shows how to associate the virtual routing interface group with a VLAN group.

```
device(config)# vlan-group 1
device(config-vlan-group-1)# group-router-interface
device(config-vlan-group-1)# exit
device(config)# interface group-ve 1
```

# interface lag

Configures LAG virtual interface that represents the entire LAG.

## Syntax

**interface lag** { *lag-interface-id* [ **to** *lag-interface-id* | [ **lag** *lag-interface-id* **to** *lag-interface-id* | **lag** *lag-interface-id* ]...] }

**no interface lag** { *lag-interface-id* [ **to** *lag-interface-id* | [ **lag** *lag-interface-id* **to** *lag-interface-id* | **lag** *lag-interface-id* ]...] }

## Command Default

LAG virtual interface is not configured.

## Parameters

*lag*

Specifies the LAG virtual interface.

*lag-interface-id*

Specifies a LAG virtual interface ID.

**to**

Specifies the range of LAG virtual interface IDs

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the configurations on the LAG virtual interface.

## Examples

The following example configures LAG virtual interface for a static LAG.

```
device(config)# lag blue static id 11
device(config-lag-blue)# ports ethernet 1/1/1 ethernet 1/1/5
device(config-lag-blue)# exit
device(config)# interface lag 11
device(config-lag-if-lg11)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# interface loopback

Configures a loopback interface and enters loopback interface configuration mode.

## Syntax

**interface loopback** *port-number*

**interface loopback** *port-number*

## Command Default

A loopback interface is not configured.

## Parameters

*port-number*

Specifies the port number for the loopback interface. The range is 1 through 32.

## Modes

Global configuration mode

## Usage Guidelines

A loopback interface adds stability to the network by working around route flap problems that can occur due to unstable links between the device and neighbors.

Loopback interfaces are always up, regardless of the states of physical interfaces.

The **no** form of the command removes the specified loopback interface.

## Examples

The following example configures a loopback interface and enters loopback interface configuration mode.

```
device(config)# interface loopback 10
device(config-lbif-3)#
```

# interface management

Specifies a management interface and enters management interface configuration mode.

## Syntax

**interface management** { **1** }

**no interface management**

## Command Default

No management interface is specified.

## Parameters

**1**

The only available interface for management.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the management interface.

## Examples

To specify the management interface and enter management interface configuration mode:

```
device# configure terminal
device(config)# interface management 1
device(config-if-mgmt-1)#
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# interface tunnel

Configures a tunnel interface.

## Syntax

**interface tunnel** *tunnel-number*

**no interface tunnel** *tunnel-number*

## Command Default

No tunnel interface is configured.

## Parameters

*tunnel-number*
Specifies the tunnel number.

## Modes

Global configuration mode

## Usage Guidelines

ICX 7150 devices do not support tunnels.

The **no** form of the command removes the tunnel interface.

## Examples

The following example creates a tunnel interface.

```
device# configure terminal
device(config)# interface tunnel 2
device(config-tnif-2)#
```

## Related Commands

tunnel destination, tunnel mode gre ip, tunnel source

# interface ve

Configures a virtual Ethernet (VE) interface.

## Syntax

**interface ve** *vlan-num*

**no interface ve** *vlan-num*

## Command Default

A virtual Ethernet interface is not configured.

## Parameters

*vlan-num*

Specifies the corresponding VLAN interface that must already be configured before the VE interface can be created. Valid values are from 1 through 4095.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

A virtual interface is a logical port associated with a Layer 3 Virtual LAN (VLAN) configured on a Layer 3 switch. You can configure routing parameters on the virtual interface to enable the Layer 3 switch to route protocol traffic from one Layer 3 VLAN to the other, without using an external router.

The no form of the command removes the VE interface.

## Examples

The following example configures a VE interface.

```
device(config)# interface ve 10
device(config-vif-10)#
```

# ip access-group

Applies numbered or named IPv4 access control lists (ACLs) to traffic entering or exiting an interface.

## Syntax

**ip access-group** { *acl-num* | *acl-name* } { **in** | **out** }

**no ip access-group** { *acl-num* | *acl-name* } { **in** | **out** }

**ip access-group** { *acl-num* | *acl-name* } **in** [ **ethernet** *unit* **/** *slot* **/** *port* ... ] [ **ethernet** *unit* **/** *slot* **/** *port* **to ethernet** *unit* **/** *slot* **/** *port* ... ]

**no ip access-group** { *acl-num* | *acl-name* } **in** [ **ethernet** *unit* **/** *slot* **/** *port* ... ] [ **ethernet** *unit* **/** *slot* **/** *port* **to ethernet** *unit* **/** *slot* **/** *port* ... ]

**ip access-group frag deny**

**no ip access-group frag deny**

## Command Default

ACLs are not applied to interfaces.

## Parameters

*acl-num*
: Specifies an ACL number. You can specify from 1 through 99 for standard ACLs and from 100 through 199 for extended ACLs.

*acl-name*
: Specifies a valid ACL name.

**in**
: Applies the ACL to inbound traffic on the port.

**out**
: Applies the ACL to outbound traffic on the port.

**ethernet** *unit* **/** *slot* **/** *port*
: Specifies the Ethernet interface from which the packets are coming.

**to ethernet** *unit* **/** *slot* **/** *port*
: Specifies the range of Ethernet interfaces from which the packets are coming.

**frag deny**
: Denies all IP fragments on the port.

## Modes

Interface subtype configuration modes

# Usage Guidelines

To apply an IPv4 ACL name that contains spaces, enclose the name in quotation marks (for example, **ip access-group "ACL for Net1" in**).

Through a virtual routing interface, you have the following options:

- (Default) Apply an ACL to all ports of the VLAN.
- One or both of the following options:
  - Apply an ACL to specified ports.
  - Apply an ACL to one or more ranges of ports.

To remove an ACL from an interface, use one of the **no** forms of this command.

# Examples

The following example creates a named standard IPv4 ACL, defines rules in the ACL, and applies it to inbound traffic on an Ethernet interface.

```
device# configure terminal
device(config)# ip access-list standard Net1
device(config-std-nacl)# deny host 10.157.22.26
device(config-std-nacl)# deny 10.157.29.12
device(config-std-nacl)# deny host IPHost1
device(config-std-nacl)# permit any
device(config-std-nacl)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# ip access-group Net1 in
```

The following example creates a named extended IPv4 ACL, defines rules in the ACL, and applies it to inbound traffic on an Ethernet interface.

```
device# configure terminal
device(config)# ip access-list extended "block Telnet"
device(config-ext-nacl-block telnet)# deny tcp host 10.157.22.26 any eq telnet
device(config-ext-nacl-block telnet)# permit ip any any
device(config-ext-nacl-block telnet)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# ip access-group "block Telnet" in
```

The first phase of the following example configures port-based VLAN 10, adds ports 1/1 through 2/12 to the VLAN, and then adds virtual routing interface 1 to the VLAN.

```
device# configure terminal
device(config)# vlan 10 name IP-subnet-vlan
device(config-vlan-10)# untag ethernet 1/1/1 to 1/1/20 ethernet 1/2/1 to 1/2/12
device(config-vlan-10)# router-interface ve 1
device(config-vlan-10)# exit
```

The next commands configure a standard numbered IPv4 ACL and define rules in it.

```
device(config)# ip access-list standard 1
device(config-std-nacl)# deny host 10.157.22.26
device(config-std-nacl)# deny 10.157.29.12
device(config-std-nacl)# deny host IPHost1
device(config-std-nacl)# permit any
```

The concluding commands apply the ACL, inbound, to a subset of the ports associated with virtual interface 1 and to outgoing traffic on all ports.

```
device(config)# interface ve 1
device(config-vif-1)# ip access-group 1 in ethernet 1/1/1 ethernet 1/1/3 ethernet 1/2/1 to 1/2/4
device(config-vif-1)# ip access-group 1 out
```

# ip access-list

Creates a named or numbered IPv4 standard or extended access control list (ACL) that permits or denies network traffic based on criteria that you specify.

## Syntax

**ip access-list** { **standard** | **extended** } { *acl-num* | *acl-name* }

**no ip access-list** { **standard** | **extended** } { *acl-num* | *acl-name* }

## Command Default

No named or numbered IPv4 ACLs are defined.

## Parameters

**standard**

Creates a standard access control list. Contains rules that permit or deny traffic based on source addresses that you specify. The rules are applicable to all ports of the specified address.

**extended**

Contains rules that permit or deny traffic according to source and destination addresses, as well as other parameters. For example, you can also filter by port, protocol (TCP or UDP), and TCP flags.

*acl-num*

Specifies the ACL number for a standard or extended access list. The value can be from 1 through 99 for standard IPv4 ACLs and from 100 through 199 for extended IPv4 ACLs.

*acl-name*

Specifies a unique IPv4 ACL name. The name can be up to 255 characters, and must begin with an alphabetic character. If the name contains spaces, put it within quotation marks. Otherwise, no special characters are allowed, except for underscores and hyphens.

## Modes

Global configuration mode

## Usage Guidelines

From Fi 08.0.80, you cannot create numbered IPv4 ACLs, using the **access-list** command. You must use this command instead.

An ACL name must be unique among IPv4 and IPv6 standard and extended ACL types.

After you create an IPv4 ACL, enter one or more **permit** or **deny** commands to create filtering rules for that ACL.

An IPv4 ACL starts functioning only after if is applied to an interface using the **ip access-group** command.

The system supports the following IPv4 ACL resources:

- IPv4 numbered standard ACLs: 99

- IPv4 numbered extended ACLs: 100
- IPv4 named standard ACLs: 99
- IPv4 named extended ACLs: 100
- Maximum filter-rules per IPv4 or IPv6 ACL: 2000. You can change the maximum up to 8192 using the **system-max ip-filter-sys** command.

The wildcard mask is in dotted-decimal notation (IP address format). It is a four-part value, where each part is 8 bits (one byte) separated by dots, and each bit is a one or a zero. Each part is a number ranging from 0 to 255, for example, 0.0.0.255. Zeros in the mask mean the packet source address must match the source IP address. Ones mean any value matches. For example, the source IP address and wildcard values 10.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 10.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in CIDR format, you can enter a forward slash (/) after the IP address, and then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of 10.157.22.26 0.0.0.255 as 10.157.22.26/24. The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant bits of the IP address into ones. For example, if you specify 10.157.22.26/24 or 10.157.22.26 0.0.0.255, and then save the changes to the startup-config file, the value appears as 10.157.22.0/24 (if you have enabled display of subnet lengths) or 10.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in "/mask-bits" format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global configuration level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

The **no** form of the command deletes the ACL. You can delete an IPv4 ACL only after you first remove it from all interfaces to which it is applied, using the **no ip access-group** command.

# Examples

The following example creates an extended, named IPv4 ACL, defines rules in it, and applies it to inbound traffic on an Ethernet interface.

```
device(config)# ip access-list extended "block Telnet"
device(config-ext-nacl-block telnet)# deny tcp host 10.157.22.26 any eq telnet
device(config-ext-nacl-block telnet)# permit ip any any
device(config-ext-nacl-block telnet)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# ip access-group "block Telnet" in
```

The following example creates an extended, numbered IPv4 ACL and defines rules in it.

```
device# configure terminal
device(config)# ip access-list extended 101
device(config-ext-nacl)# seq 30 deny udp 19.1.2.0 0.0.0.255 eq 2023 20.1.2.0 0.0.0.255 eq 2025 dscp-
mapping 23
device(config-ext-nacl)# permit 12 host 098.096.31.10 any
device(config-ext-nacl)# deny tcp  host 098.092.12.10 131.21.12.0/24 syn
device(config-ext-nacl)# deny 120  host 18.192.112.110 13.2.2.0/24 log
device(config-ext-nacl)# permit ip any any mirror
```

The following example configures a standard ACL.

```
device# configure terminal
device(config)# ip access-list standard acl1
device(config-std-nacl)#
```

The following example shows how to configure a standard ACL.

```
device# configure terminal
device(config)# ip access-list standard 1
device(config-std-nacl)# deny host 10.157.22.26 log
device(config-std-nacl)# deny 10.157.29.12 log
device(config-std-nacl)# deny host IPHost1 log
device(config-std-nacl)# permit any
device(config-std-nacl)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip access-group 1 in
device(config)# write memory
```

The following example configures an extended ACL.

```
device# configure terminal
device(config)# ip access-list extended 125
device(config-ext-nacl)#
```

The following example deletes an IPv4 ACL.

```
device# configure terminal
device(config)# no ip access-list standard acl1
```

# ip address

Configures an IP address on an interface.

## Syntax

**ip address** *ip-address/mask* { **dynamic** | **ospf-ignore** | **ospf-passive** } [ **replace** ]

**ip address** *ip-address/mask* [ **secondary** ]

**no ip address** [ *ip-address/mask* ]

## Parameters

*ip-address*
Specifies the IP address.

*mask*
IP address or subnet mask length.

**dynamic**
Specifies the interface IP address is dynamic.

**ospf-ignore**
Disables adjacency formation with OSPF neighbors and disables advertisement of the interface to OSPF.

**ospf-passive**
Disables adjacency formation with OSPF neighbors but does not disable advertisement of the interface to OSPF.

**replace**
Replaces the configured primary IP address on the interface.

**secondary**
Specifies that the configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.

## Modes

Interface configuration mode

Management interface configuration mode

## Usage Guidelines

- Use this command to configure a primary or secondary IP address for a specific interface. You can also use this command to prevent OSPF from running on specified subnets. Multiple primary IP addresses are supported on an interface.
- You can use this command to configure a primary or secondary IP address for a management interface.
- For a management interface, only one primary IP address is supported. Secondary IP addresses are not supported.
- A primary IP address cannot overlap with a previously configured IP subnet.
- A primary IP address must be configured before you configure a secondary IP address in the same subnet.

- To remove the configured static or DHCP address, enter **no ip address**. This resets the address to 0.0.0.0/0.
- The **no** form of the command removes a specific IP address from the interface.

# Examples

The following example configures a primary IP address on a specified Ethernet interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip address 10.1.1.1/24
```

The following example replaces the primary IP address of an interface. .

```
device# configure terminal
device (config)# interface ethernet 1/1/21
device(config-if-e1000-1/1/21)# ip address 10.1.1.2/24 replace
```

# ip-address

Configures a virtual IP address for a Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRP-E) instance.

## Syntax

**ip-address** *ip-address*

**no ip-address** *ip-address*

## Command Default

A virtual IP address is not configured for a VRRP or VRRP-E instance.

## Parameters

*ip-address*

Configures the IP address, in dotted-decimal format.

## Modes

VRID interface configuration mode

## Usage Guidelines

For VRRP instances, the IP address used for the virtual router must be configured on the device assigned to be the initial VRRP owner device. The same IP address cannot be used on any other VRRP device.

For VRRP-E instances, the IP address used for the virtual router must not be configured on any other device.

The **no** form of this command removes the virtual router IP address.

## Examples

The following example configures a virtual IP address for VRID 1 when VRRP is implemented. In this example, the device is configured as the VRRP owner device.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.5.1
device(config-if-e1000-1/1/6-vrid-1)# activate
VRRP router 1 for this interface is activating
```

The following example configures a virtual IP address for VRID 2 when VRRP-E is implemented. In this example, the device is configured as a VRRP backup device and the highest priority device will become the master VRRP device.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.5.3/24
device(config-if-e1000-1/1/5)# ip vrrp-extended vrid 2
device(config-if-e1000-1/1/5-vrid-2)# backup priority 110
device(config-if-e1000-1/1/5-vrid-2)# version 2
device(config-if-e1000-1/1/5-vrid-2)# ip-address 10.53.5.254
device(config-if-e1000-1/1/5-vrid-2)# activate
VRRP router 2 for this interface is activating
```

# ip-address (VSRP)

Configures the IP address to back up.

## Syntax

**ip-address** *ip-address*

**no ip-address** *ip-address*

**ip address** *ip-address*

**no ip address** *ip-address*

## Command Default

The IP address to backup is not configured.

## Parameters

*ip-address*

Configures the IP address to back up.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

If you are configuring a Layer 3 switch for VSRP, you can specify an IP address to back up. When you specify an IP address, VSRP provides redundancy for the address. This is useful if you want to back up the gateway address used by hosts attached to the VSRP backups. VSRP does not require you to specify an IP address. If you do not specify an IP address, VSRP provides Layer 2 redundancy. If you do specify an IP address, VSRP provides Layer 2 and Layer 3 redundancy.

The VRID IP address must be in the same subnet as a real IP address configured on the VSRP interface, but cannot be the same as a real IP address configured on the interface.

Failover applies to both Layer 2 and Layer 3.

The **no** form of the command removes the configured backup IP address.

## Examples

The following example configures the backup IP address.

```
device(config)# vlan 200
device(config-vlan-200)# tagged ethernet 1/1/1 to 1/1/8
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# ip-address 10.10.10.1
```

# ip arp inspection syslog disable

Disables the syslog messages for Dynamic ARP Inspection.

## Syntax

**ip arp inspection syslog disable**

**no ip arp inspection syslog disable**

## Command Default

Syslog messages are enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command re-enables syslog messages for Dynamic ARP Inspection.

## Examples

The following example disables the syslog messages for dynamic ARP inspection.

```
device(config)# ip arp inspection syslog disable
```

## History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was introduced. |

# ip arp inspection validate

Enables validation of the ARP packet destination MAC, ARP Packet IP, and source MAC addresses.

## Syntax

**ip arp inspection validate** [**dst-mac** | **ip** | **src-mac**]

## Command Default

IP ARP packet destination address validation is disabled.

## Parameters

**dst-mac**

Checks the destination MAC address in the Ethernet header against the target MAC address in the ARP body for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped.

**ip**

Checks the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses.

**src-mac**

Checks the source MAC address in the Ethernet header against the sender MAC address in the ARP body for ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped.

## Modes

Global configuration mode

## Usage Guidelines

You can enable validation of ARP packet destination addresses for a single destination address or for all destination addresses.

You must execute the command once for each type of ARP packet destination address you want to validate.

## Examples

The following example enables validation of the MAC, ARP Packet IP, and source MAC ARP packet destination addresses.

```
device(config)# configure terminal
device(config)# ip arp inspection validate dst-mac
device(config)# ip arp inspection validate src-mac
device(config)# ip arp inspection validate ip
```

# History

| Release version | Command history |
|---|---|
| 08.0.10a | This command was introduced. |

# ip arp inspection vlan

Enables dynamic ARP inspection (DAI) on a VLAN or a range of VLANs.

## Syntax

**ip arp inspection vlan** *vlan-id* [ **to** *vlan-id* ... ]

**no ip arp inspection vlan** *vlan-id* [ **to** *vlan-id* ... ]

## Command Default

Dynamic ARP inspection is disabled by default.

## Parameters

*vlan-id*

Specifies the VLAN number.

**to** *vlan-id*

Specifies a range of VLANs.

## Modes

Global configuration mode

## Usage Guidelines

All VLANs included in the range when using the **to** keyword must be valid VLANs. Otherwise an error will occur.

The maximum number of VLANS that can be configured using the **to** keyword is 1024.

The **no** form of the command disables DAI on the VLAN.

## Examples

The following example enables DAI on VLAN 2.

```
device# configure terminal
device(config)# ip arp inspection vlan 2
```

The following example enables DAI on VLANs 100 through 150, VLAN 160, and VLANs 170 through 200.

```
device# configure terminal
device(config)# ip arp inspection vlan 100 to 150 160 170 to 200
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | The **to** keyword was added to enable Dynamic ARP Inspection on a range of VLANs by using a single command.. |

# ip arp learn-gratuitous-arp

Enables learning gratuitous ARP.

## Syntax

**ip arp learn-gratuitous-arp**

**no ip arp learn-gratuitous-arp**

## Command Default

Learning gratuitous ARP is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

Learning gratuitous ARP enables Layer 3 devices to learn ARP entries from incoming gratuitous ARP packets from the hosts which are directly connected. This help achieve faster convergence for the hosts when they are ready to send traffic.

A new ARP entry is created when a gratuitous ARP packet is received. If the ARP is already existing, it will be updated with the new content.

The **no** form of the command disables learning gratuitous ARP.

## Examples

The following example enables learning gratuitous ARP.

```
device(config)# ip arp learn-gratuitous-arp
```

# ip arp port-move-syslog

Disables or re-enables Address Resolution Protocol (ARP) port movement syslog message generation.

## Syntax

**ip arp port-move-syslog**

**no ip arp port-move-syslog**

## Command Default

Syslog message is generated with every port movement for ARP entries by default.

## Modes

Global configuration mode

## Usage Guidelines

Whenever a port, on which a MAC address for an ARP is learned, is moved to a different port, a syslog message is generated by default. This may cause flooding of the syslog server or console with syslog messages in certain deployments where next hop or ARP port movement occurs continuously. In such scenarios, the default behavior can be disabled and syslog messages can be prevented from being generated with every port movement for ARP entries using the **no ip arp port-move-syslog** command.

The **no** form of the command disables ARP port movement syslog generation.

## Examples

The following example disables ARP port movement syslog message generation.

```
device(config)# no ip arp port-move-syslog
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ip arp-age

Configures ARP aging parameter.

## Syntax

**ip arp-age** *age-time*

**no ip arp-age** *age-time*

## Command Default

The default ARP aging is 10 minutes.

## Parameters

*age-time*

Specifies the ARP age time in minutes. Valid range is from 0 to 240, 0 disables aging. The default is 10 minutes.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

When the Layer 3 switch places an entry in the ARP cache, the Layer 3 switch also starts an aging timer for the entry. The aging timer ensures that the ARP cache does not retain learned entries that are no longer valid. An entry can become invalid when the device with the MAC address of the entry is no longer on the network.

The ARP age affects dynamic (learned) entries only, not static entries. The default ARP age is ten minutes. On Layer 3 switches, you can change the ARP age to a value from 0 through 240 minutes. You cannot change the ARP age on Layer 2 switches. If you set the ARP age to zero, aging is disabled and entries do not age out.

Use the command from interface configuration mode to override the globally configured IP ARP age on an individual interface.

The **no** form of the command resets the ARP aging to the default value of 10 minutes.

## Examples

The following example configures the ARP aging time as 100 minutes.

```
device(config)# ip arp-age 100
```

The following example overrides the global ARP aging time on a particular interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip arp-age 30
```

# ip bootp-gateway

Changes the IP address used for stamping BootP or DHCP requests received on the interface.

## Syntax

**ip bootp-gateway** *ip-address*

## Parameters

*ip-address*

Specifies the IP address used to stamp requests received on the interface.

## Modes

Interface configuration mode

## Usage Guidelines

The BootP or DHCP stamp address is an interface parameter. Use this command to change the parameter on the interface that is connected to the BootP/DHCP client.

In the example given below the command changes the CLI to the configuration level for port 1/1/1, then changes the BootP or DHCP stamp address for requests received on port 1/1/1 to 10.157.22.26. The Layer 3 switch will place this IP address in the Gateway Address field of BootP or DHCP requests that the Layer 3 switch receives on port 1/1/1 and forwards to the BootP or DHCP server.

## Examples

The following command changes the IP address used for stamping BootP or DHCP requests received on interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-1/1/1)# ip bootp-gateway 10.157.22.26
```

# ip bootp-use-intf-ip

Configures a Dynamic Host Configuration Protocol (DHCP) relay agent to set the source IP address of a DHCP-client packet with the IP address of the interface in which the DHCP-client packet is received.

## Syntax

**ip bootp-use-intf-ip**

**no ip bootp-use-intf-ip**

## Command Default

The DHCP relay agent sets the source IP address of a DHCP-client packet with the IP address of the outgoing interface to the DHCP server.

## Modes

Global configuration mode

## Usage Guidelines

You can configure ACLs on a DHCP server to permit or block access to the DHCP server from particular subnets or networks. You can then use this command on the DHCP relay agent to reveal the source subnet or network of a DHCP packet to the DHCP server, which enables the DHCP server to process or discard the DHCP traffic according to the configured ACLs.

The **no** form of the command restores the default behavior. The DHCP relay agent sets the source IP address of a DHCP-client packet with the IP address of the outgoing interface to the DHCP server.

## Examples

The following example configures a FastIron DHCP relay agent so that it sets the source IP address of a DHCP-client packet with the IP address of the interface on which the DHCP-client packet is received.

```
device(config)# ip bootp-use-intf-ip
```

# ip broadcast-zero

Enables the Layer 3 switch for zero-based IP subnet broadcasts in addition to ones-based IP subnet broadcasts.

## Syntax

**ip broadcast-zero**

**no ip broadcast-zero**

## Command Default

By default, the Layer 3 switch treats IP packets with all ones in the host portion of the address as IP broadcast packets.

## Modes

Global configuration mode

## Usage Guidelines

Most IP hosts are configured to receive IP subnet broadcast packets with all ones in the host portion of the address. However, some older IP hosts instead expect IP subnet broadcast packets that have all zeros instead of all ones in the host portion of the address. To accommodate this type of host, you can enable the Layer 3 switch to treat IP packets with all zeros in the host portion of the destination IP address as broadcast packets.

When you enable the Layer 3 switch for zero-based subnet broadcasts, the Layer 3 switch still treats IP packets with all ones the host portion as IP subnet broadcasts too. Thus, the Layer 3 switch can be configured to support all ones only (the default) or all ones and all zeroes.

**NOTE**
This feature applies only to IP subnet broadcasts, not to local network broadcasts. The local network broadcast address is still expected to be all ones.

You must save the configuration and reload the software to place this configuration change into effect.

The **no** form of the command disablesthe Layer 3 switch for zero-based IP subnet broadcasts in addition to ones-based IP subnet broadcasts.

## Examples

The following example enables the Layer 3 switch for zero-based IP subnet broadcasts in addition to ones-based IP subnet broadcasts.

```
device(config)# ip broadcast-zero
device(config)# write memory
device(config)# end
device# reload
```

# ip default-gateway

Configures the default gateway for a Layer 2 switch.

## Syntax

**ip default-gateway** *ip-address*

**no ip default-gateway** *ip-address*

## Command Default

Default gateway is not configured.

## Parameters

*ip-address*

Specifies the IP address of the default gateway.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the configured default gateway.

## Examples

The following example configures the default gateway.

```
device(config)# ip default-gateway 10.30.5.1
```

# ip default-network

Configures a default network route.

## Syntax

**ip default-network** *ip-address*

**no ip default-network** *ip-address*

## Command Default

Default network is not configured.

## Parameters

*ip-address*

Specifies the IP address of the network in the format A.B.C.D/L or A.B.C.D followed by the network mask in dotted decimal format.

## Modes

Global configuration mode

## Usage Guidelines

The Layer 3 switch enables you to specify a candidate default route without the need to specify the next hop gateway. If the IP route table does not contain an explicit default route (for example, 0.0.0.0/0) or propagate an explicit default route through routing protocols, the software can use the default network route as a default route instead.

When the software uses the default network route, it also uses the default network route's next hop gateway as the gateway of last resort. Configuring the default network route is especially useful in environments where network topology changes can make the next hop gateway unreachable. This feature allows the Layer 3 switch to perform default routing even if the default network route's default gateway changes.

You can configure up to four default network routes.

The **no** form of the command removes the default network route.

## Examples

The following example configures a default IP network route.

```
device(config)# ip default-network 10.157.22.0
device(config)# write memory
```

# ip dhcp-client auto-update enable

Enables the DHCP auto-update functionality.

## Syntax

**ip dhcp-client auto-update enable**

**no ip dhcp-client auto-update enable**

## Command Default

DHCP client auto-update is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables DHCP auto-update.

## Examples

The following example re-enables auto-update.

```
device(config)# ip dhcp-client auto-update enable
```

# ip dhcp-client enable

Enables DHCP client auto-update.

## Syntax

**ip dhcp-client enable**

**no ip dhcp-client enable**

## Modes

Global configuration mode.

Interface configuration mode.

## Usage Guidelines

You can enable this command on a switch in global configuration mode. On routers, you can enable this command in interface configuration mode.

The **no** form of the command disables the DHCP client.

## Examples

The following example enables the DHCP client on a switch.

```
device(config)# ip dhcp-client enable
```

The following example enables the DHCP client on a router.

```
device(config-if-e1000-1/1/1)# ip dhcp-client enable
```

On a router, enter the **ip dhcp-client enable** command to re-enable the DHCP client.

```
device(config-if-ve1)# ip dhcp-client enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | Example for enabling DHCP client on VE was added. |

# ip dhcp-server arp-ping-timeout

Sets the ARP-ping timeout value.

## Syntax

**ip dhcp-server arp-ping-timeout** *number*

**no ip dhcp-server arp-ping-timeout** *number*

## Command Default

ARP-ping timeout is not enabled.

## Parameters

*number*

The number of seconds to wait for a response to an ARP-ping packet. The minimum setting is 5 seconds and the maximum is 30 seconds.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command disables the ARP ping timeout. If there is no response to the ARP-ping packet within a set amount of time (set in seconds), the server deletes the client from the lease-binding database.

**NOTE**
Do not alter the default value unless it is necessary. Increasing the value of this timer may increase the time to get console access after a reboot.

## Examples

The following example sets the ARP-ping timeout to 25 seconds.

```
device# ip dhcp-server arp-ping-timeout 25
```

# ip dhcp-server enable

Enables the DHCP server.

## Syntax

**ip dhcp-server enable**

**no ip dhcp-server enable**

## Command Default

The DHCP server is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command disables the DHCP server.

## Examples

The following example enables the DHCP server.

```
device(config)# ip dhcp-server enable
```

# ip dhcp-server mgmt

Enables or disables the DHCP server on the management port.

## Syntax

**ip dhcp-server mgmt**

**no ip dhcp-server mgmt**

## Command Default

DHCP server management is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables the DHCP server on the management port.

When disabled, DHCP client requests that are received on the management port are discarded.

## Examples

The following example enables the DHCP server on the management port.

```
device(config)# ip dhcp-server mgmt
```

The following example disables the DHCP server on the management port.

```
device(config)# no ip dhcp-server mgmt
```

# ip dhcp-server pool

Creates a DHCP server address pool.

## Syntax

**ip dhcp-server pool** *name*

**no ip dhcp-server pool** *name*

## Parameters

*name*

The name of the address pool.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of this command disables the address pool. Use this command to switch to pool configuration mode (config-dhcp-name# prompt) and create an address pool.

## Examples

The following example creates a DHCP address pool.

```
device(config)# ip dhcp-server pool cabo
```

# ip dhcp-server relay-agent-echo enable

Activates the DHCP option 82.

## Syntax

**ip dhcp-server relay-agent-echo enable**

## Command Default

The DHCP option 82 functionality is not enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

This command enables the DHCP server to echo the entire contents of the relay agent information option in all replies.

## Examples

The following example enables the DHCP server relay agent.

```
device(config)# ip dhcp-server relay-agent-echo enable
```

# ip dhcp-server server-identifier

Specifies the IP address of the selected DHCP server.

## Syntax

**ip dhcp-server server-identifier** *ip-address*

## Parameters

*ip-address*

Specifies the IP address of the DHCP server.

## Modes

Global configuration mode

## Examples

The following example shows assigning an IP address to the selected DHCP server.

```
device(config)# ip dhcp-server-identifier 10.1.1.144
```

# ip dhcp snooping relay information disable

Enables DHCP snooping relay information (DHCP Option 82) on a specified VLAN or for all VLANs.

## Syntax

**ip dhcp snooping relay information disable** [ **vlan** *vlan-id* ]

**no ip dhcp snooping relay information disable** [ **vlan** *vlan-id* ]

## Command Default

DHCP option 82 is enabled by default when DHCP snooping is enabled.

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

## Modes

Global configuration mode

## Usage Guidelines

When DHCP snooping is enabled using the **ip dhcp snooping vlan** command, DHCP option 82 is automatically enabled. If multiple interfaces are part of a VLAN, DHCP option 82 can be disabled, or re-enabled, for all ports of the VLAN using the **ip dhcp snooping relay information disable** command. DHCP option 82 can also be disabled or re-enabled for all VLANs in a switch using the **ip dhcp snooping relay information disable** command.

The **no** form of the command re-enables DHCP option 82 on a specified VLAN or for all VLANs.

## Examples

The following example disables DHCP option 82 globally for all VLANs so that it is not enabled when DHCP snooping is configured for VLAN 100.

```
device# configure terminal
device(config)# ip dhcp snooping relay information disable
device(config)# ip dhcp snooping vlan 100
```

The following example disables DHCP option 82 on all ports globally and re-enables it for interface Ethernet 1/1/1.

```
device# configure terminal
device(config)# ip dhcp snooping relay information disable
device(config)# ip dhcp snooping vlan 100
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# dhcp snooping relay information
```

The following example disables DHCP option 82 globally and enables IP DHCP snooping for VLANs 100, 200, and 300. DHCP option 82 is then enabled on all ports for VLAN 100.

```
device# configure terminal
device(config)# ip dhcp snooping relay information disable
device(config)# ip dhcp snooping vlan 100
device(config)# ip dhcp snooping vlan 200
device(config)# ip dhcp snooping vlan 300
device(config)# no ip dhcp snooping relay information disable vlan 100
```

The following example disables DHCP option 82 for VLAN 100. DHCP option 82 is automatically configured for VLANs 200 and 300 when DHCP snooping is enabled.

```
device# configure terminal
device(config)# ip dhcp snooping vlan 100
device(config)# ip dhcp snooping vlan 200
device(config)# ip dhcp snooping vlan 300
device(config)# ip dhcp snooping relay information disable vlan 100
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# ip dhcp snooping vlan

Enables DHCP snooping on a VLAN or a range of VLANs.

## Syntax

**ip dhcp snooping vlan** *vlan-id* [ **to** *vlan-id* ... ]

**no ip dhcp snooping vlan** *vlan-id* [ **to** *vlan-id* ... ]

## Command Default

DHCP snooping is disabled by default.

## Parameters

*vlan-id*

Specifies the ID of a configured client or DHCP server VLAN.

**to** *vlan-id*

Specifies a range of VLANs.

## Modes

Global configuration mode

## Usage Guidelines

When DHCP snooping is enabled on a VLAN, DHCP packets are inspected. DHCP snooping must be enabled on the client and the DHCP server VLANs.

All VLANs included in the range when using the **to** keyword must be valid VLANs. Otherwise an error will occur.

The **no** form of the command disables DHCP snooping on the specified VLAN.

## Examples

The following example enables DHCP snooping on VLAN 2.

```
device# configure terminal
device(config)# ip dhcp snooping vlan 2
```

The following example configures VLANs 100 through 150, VLAN 160, and VLANs 170 through 200 and enables DHCP snooping on all of the configured VLANs.

```
device# configure terminal
device(config)# vlan 100 to 150
device(config-mvlan-100-150)# tagged ethernet 1/1/12
device(config-mvlan-100-150)# exit
device(config)# vlan 151 to 200
device(config-mvlan-151-200)# tagged ethernet 1/1/12
device(config-mvlan-100-150)# exit
device(config)# ip dhcp snooping vlan 100 to 150 160 170 to 200
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | The **to** keyword was added to enable DHCP snooping on a range of VLANs by using a single command. |

# ip dhcp relay information policy

Configures the DHCP relay information policy.

## Syntax

**ip dhcp relay information policy** [**drop** | **keep** | **replace**]

## Command Default

The device replaces the information with its own relay agent information.

## Parameters

**drop**

Configures the device to discard messages containing relay agent information.

**keep**

Configures the device to keep the existing relay agent information.

**replace**

Configures the device to overwrite the relay agent information with the information in the configuration.

## Modes

Global configuration mode.

## Usage Guidelines

When the device receives a DHCP message that contains relay agent information, if desired, you can configure the device to keep the information instead of replacing it, or to drop (discard) messages that contain relay agent information.

## Examples

The following example configures the device to keep the relay agent information contained in a DHCP message.

```
device(config)# ip dhcp relay information policy keep
```

The following example configures the device to drop the relay agent information contained in a DHCP message.

```
device(config)# ip dhcp relay information policy drop
```

# ip directed-broadcast

Enables directed broadcast forwarding.

## Syntax

**ip directed-broadcast**

**no ip directed-broadcast**

## Command Default

Directed broadcast forwarding is disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

A Smurf attack relies on the intermediary to broadcast ICMP echo request packets to hosts on a target subnet. When the ICMP echo request packet arrives at the target subnet, it is converted to a Layer 2 broadcast and sent to the connected hosts. This conversion takes place only when directed broadcast forwarding is enabled on the device. To avoid being an intermediary in a Smurf attack, make sure forwarding of directed broadcasts is disabled on the device.

The **no** form of the command disables directed broadcast forwarding.

## Examples

The following example enables directed broadcast forwarding.

```
device(config)# ip directed-broadcast
```

# ip dns

Configures the IPv4 Domain Name System (DNS).

## Syntax

**ip dns** { **domain-list** *domain-name* | **server-address** *ip-address* [ *ip-address...* ] }

**no ip dns** { **domain-list** *domain-name* | **server-address** *ip-address* [ *ip-address...* ] }

## Command Default

IP DNS is not configured.

## Parameters

**domain-list**
Configures a list of DNS domains.

*domain-name*
The domain name.

**server-address**
Configures the DNS server IPv4 address.

*ip-address*
The IPv4 address of the DNS server. You can configure up to a maximum of four IP addresses separated by a space.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the DNS configurations.

## Examples

The following example shows how to configure an IPv4 address for a DNS server.

```
device(config)# ip dns server-address 192.168.10.1 192.168.100.1
```

The following example shows how to configure the DNS domain-list.

```
device(config)# ip dns domain-list company.com
```

# ip dscp-remark

Enables remarking of the differentiated services code point (DSCP) field for all IPv4 packets.

## Syntax

**ip dscp-remark** *dscp-value*

**no ip dscp-remark** *dscp-value*

## Command Default

DSCP remarking is disabled.

## Parameters

*dscp-value*

Specifies the DSCP value ranges you are remarking.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of this command disables DSCP remarking.

In interface configuration mode, the command enables DSCP remarking for the given port. The configuration can be done on a physical port, LAG, and VE port.

If DHCP snooping is enabled, you cannot globally enable DSCP remarking. When you enter the global configuration **ip dscp-remark** command, the following error message is displayed.

```
Error: DHCP Snooping is configured on the system. Cannot enable DSCP remarking
```

## Examples

The following example globally enables DSCP remarking on all IPv4 packets when the DSCP bit value is 40:

```
Device(config)# ip dscp-remark 40
```

The following example enables DSCP remarking on all IPv4 packets received on a specific port when the DSCP bit value is 50:

```
Device(config)# interface ethernet1/1/1
Device(config-if-e1000-1/1/1)# ip dscp-remark 50
```

# ip encapsulation

Changes the IP encapsulation type.

## Syntax

**ip encapsulation** { **ethernet-2** | **snap** }

**no ip encapsulation** { **ethernet-2** | **snap** }

## Command Default

Layer 3 switches use Ethernet II by default.

## Parameters

**ethernet-2**

Configures the IP encapsulation type as Ethernet II.

**snap**

Configures the IP encapsulation type as Ethernet SNAP (also called IEEE 802.3).

## Modes

Interface configuration mode

## Usage Guidelines

The Layer 3 switch encapsulates IP packets into Layer 2 packets, to send the IP packets on the network.

All IP devices on an Ethernet network must use the same format. Layer 3 switches use Ethernet II by default. All devices connected to the Layer 3 switch port must use the same encapsulation type.

The **no** form of the command resets the encapsulation type as Ethernet II.

## Examples

The following example configures the IP encapsulation type as Ethernet SNAP.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip encapsulation snap
```

# ip follow ve

Configures a virtual routing interface to share the IP address with other virtual routing interfaces.

## Syntax

**ip follow ve** *number*

**no ip follow ve** *number*

## Command Default

A virtual routing interface does not share its IP address with other interfaces.

## Parameters

*number*

Specifies the virtual routing interface number.

## Modes

Virtual routing configuration mode

## Usage Guidelines

When this command is configured, one virtual routing interface is configured with an IP address, while the other virtual routing interfaces are configured to use that IP address, thus, they "follow" the virtual routing interface that has the IP address. This is helpful in conserving IP address space.

When configuring IP Follow, the primary virtual routing interface should not have ACL or DoS Protection configured. It is recommended that you create a dummy virtual routing interface as the primary and use the IP-follow virtual routing interface for the network. Global Policy Based Routing is not supported when IP Follow is configured. IPv6 is not supported with IP Follow. FastIron devices support IP Follow with OSPF and VRRP protocols only.

The **no** form of the command removes the configuration.

## Examples

The following example configures IP Follow.

```
device(config)# vlan 2 name IP-Subnet_10.1.2.0/24
device(config-vlan-2)# untag ethernet 1 to 4
device(config-vlan-2)# router-interface ve 1
device(config-vlan-2)# interface ve 1
device(config-vif-1)# ip address 10.10.2.1/24
device(config-vif-1)# interface ve 2
device(config-vif-2)# ip follow ve 1
device(config-vif-2)# interface ve 3
device(config-vif-3)# ip follow ve 1
```

# ip forward-protocol udp

Configures the Layer 3 switch to forward client requests for UDP applications.

## Syntax

**ip forward-protocol udp** { *port-name* | *port-num* }

**no ip forward-protocol udp** { *port-name* | *port-num* }

## Command Default

Layer 3 switch does not forward client requests for UDP applications.

## Parameters

*port-name*

Specifies the UDP application name. The values can be **echo**, **discard**, **time**, **tacacs**, **dns**, **bootps**, **bootpc**, **tftp**, **ntp**, **netbios-ns**, **netbios-dgm**, **mobile-ip**, and **talk**.

*port-num*

Specifies the UDP application port number.

## Modes

Global configuration mode

## Usage Guidelines

You also must configure a helper address on the interface that is connected to the clients for the application. The Layer 3 switch cannot forward the requests unless you configure the helper address

This command disables forwarding of SNMP requests to the helper addresses configured on Layer 3 switch interfaces.

The **no** form of the command stops forwarding client requests for the UDP applications.

## Examples

The following example enables the forwarding of NTP broadcasts.

```
device(config)# ip forward-protocol udp ntp
```

# ip helper-address

Configures a helper address on the interface connected to the client to enable forwarding of client broadcast request for a UDP application when the client and server are on different networks.

## Syntax

**ip helper-address** *address-number* [ *ip-address* [ **unicast** ] ]

**no ip helper-address** *address-number* [ *ip-address* [ **unicast** ] ]

## Command Default

IP helper address is not configured.

## Parameters

*address-number*
Specifies the IP helper address number. Valid values are 1 to 16.

*ip-address*
Specifies the server IP address or the subnet directed broadcast address of the IPsubnet the server is in.

**unicast**
Specifies that the client request must be forwarded to the server that is on thesame network.

## Modes

Interface configuration mode

## Usage Guidelines

To forward a client broadcast request for a UDP application when the client and server are on different networks, you must configure a helper address on the interface connected to the client. Specify the server IP address or the subnet directed broadcast address of the IP subnet the server is in as the helper address.

You can configure up to 16 helper addresses on each interface. You can configure a helper address on an Ethernet port or a virtual interface.

By default, IP helper does not forward client broadcast request to a server within the network. To forward a client broadcast request when the client and server are on the same network, configure an IP helper with **unicast** option on the interface connected to the client.

The **no** form of the command removes the configured helper address.

## Examples

The following example configures an IP helper address on Ethernet interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip helper-address 1 10.20.3.4
```

The following example configures an IP helper address on Ethernet interface 1/2/2 to enable forwarding of broadcast request to a server within the network.

```
device(config)# interface ethernet 1/2/2
device(config-if-e1000-1/1/1)# ip helper-address 2 10.10.3.4 unicast
```

# ip helper-use-responder-ip

Configures the device so that a BOOTP or DHCP reply to a client contains the server IP address as the source address instead of the router IP address.

## Syntax

**ip helper-use-responder-ip**

**no ip helper-use-responder-ip**

## Modes

Global configuration mode

## Examples

The following example retains the responder source IP in the reply.

```
device(config)# ip helper-use-responder-ip
```

# ip hitless-route-purge-timer

Configures the timer to set the duration for which the routes should be preserved after switchover.

## Syntax

**ip hitless-route-purge-timer** *seconds*

**no ip hitless-route-purge-timer** *seconds*

## Command Default

The default timer setting is 45 seconds.

## Parameters

*seconds*

Specifies the time after switchover to start IPv4 route purge. The value can range from 2 to 600 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the configured value and sets the timer to the default 45 seconds.

## Examples

The following example shows how to set the IPv4 hitless purge timer to 60 seconds.

```
device(config)# ip hitless-route-purge-timer 60
```

# ip icmp burst-normal

Configures the device to drop ICMP packets when excessive number of packets are encountered.

## Syntax

**ip icmp burst-normal** *num-packets* **burst-max** *num-packets* **lockup** *time*

**no ip icmp burst-normal** *num-packets* **burst-max** *num-packets* **lockup** *time*

**ip icmp attack-rate burst-normal** *num-packets* **burst-max** *num-packets* **lockup** *time*

**no ip icmp attack-rate burst-normal** *num-packets* **burst-max** *num-packets* **lockup** *time*

## Command Default

Threshold values for ICMP packets are configured.

## Parameters

*num-packets*

Configures the number of packets per second in normal burst mode. Valid values are from 1 through 100,000 packets per second.

**NOTE**
For the Ruckus ICX 7750, the value is in Kbps.

**burst-max** *num-packets*

Specifies the number of packets per second in maximum burst mode. Valid values are 1 through 100,000 packets per second.

**NOTE**
For the Ruckus ICX 7750, the value is in Kbps.

**lockup** *time*

Configures the lockup period in seconds. Valid values are from 1 through 10,000 seconds.

**attack-rate**

Configures the attack rate. This is specific to the Ruckus ICX 7750.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

You can configure the device to drop ICMP packets when excessive number of packets are encountered, as is the case when the device is the victim of a Smurf attack. You can set threshold values for ICMP packets that are targeted at the router itself or passing through an interface, and drop them when the thresholds are exceeded.

For Layer 3 router code, if the interface is part of a VLAN that has a router VE, you must configure ICMP attack protection at the VE level. When ICMP attack protection is configured at the VE level, it will apply to routed traffic only. It will not affect switched traffic.

**NOTE**
You must configure VLAN information for the port before configuring ICMP attack protection. You cannot change the VLAN configuration for a port on which ICMP attack protection is enabled.

The **no** form of the command removes the configured threshold value.

# Examples

The following example sets threshold values for ICMP packets targeted at the router.

```
device(config)# ip icmp burst-normal 5000 burst-max 10000 lockup 300
```

The following example sets threshold values for ICMP packets received on interface 3/1/1.

```
device(config)# interface ethernet 3/1/1
device(config-if-e1000-3/1/1)# ip icmp burst-normal 5000 burst-max 10000 lockup 300
```

The following example sets the threshold value for ICMP packets received on VE 31.

```
device(config)# interface ve 31
device(config-vif-31)# ip icmp burst-normal 5000 burst-max 10000 lockup 300
```

# ip icmp echo broadcast-request

Enables an ICMP echo response caused by a broadcast echo request.

## Syntax

**ip icmp echo broadcast-request**

**no ip icmp echo broadcast-request**

## Command Default

By default, devices are enabled to respond to broadcast ICMP echo packets, which are ping requests.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables the response to broadcast ICMP echo packets (ping requests).

## Examples

The following example enables an ICMP echo response caused by a broadcast echo request.

```
device(config)# ip icmp echo broadcast-request
```

# ip icmp redirects

Enables IPv4 ICMP redirect messages.

## Syntax

**ip icmp redirects**

**no ip icmp redirects**

## Command Default

By default, IP ICMP redirect at the global level is disabled and a Layer 3 switch does not send an ICMP redirect message to the source of a misdirected packet in addition to forwarding the packet to the appropriate router.

## Modes

Global configuration mode

VE interface configuration mode

## Usage Guidelines

You can enable and disable IPv4 ICMP redirect messages globally or on individual Virtual Ethernet (VE) interfaces, but not on individual physical interfaces.

**NOTE**
The device forwards misdirected traffic to the appropriate router, even if you disable the redirect messages.

The **no** form of the command removes the ICMP redirect control.

## Examples

The following example configures the IP redirect messages at the global level.

```
device(config)# ip icmp redirects
```

The following example configures the IP redirect messages on a VE interface.

```
device(config)# interface ve 10
device(config-vif-10)# ip icmp redirects
```

# ip icmp unreachable

Enables sending ICMP unreachable messages.

## Syntax

**ip icmp unreachable** { **administration** | **fragmentation-needed** | **host** | **network** | **port** | **protocol** | **source-route-fail**}

**no ip icmp unreachable** { **administration** | **fragmentation-needed** | **host** | **network** | **port** | **protocol** | **source-route-fail**}

## Command Default

By default, when a device receives an IP packet that the device cannot deliver, the device sends an ICMP unreachable message back to the host that sent the packet.

## Parameters

**administration**

Sends the ICMP unreachable message when the packet is dropped by the device due to a filter or ACL configured on the device.

**fragmentation-needed**

Sends the ICMP unreachable message when the packet has the Do not Fragment bit set in the IP Flag field, but the device cannot forward the packet without fragmenting it.

**host**

Sends the ICMP unreachable message when the destination network or subnet of the packet is directly connected to the device, but the host specified in the destination IP address of the packet is not on the network.

**network**

Sends the ICMP unreachable message when the destination network is

**port**

Sends the ICMP unreachable message when the destination host does not have the destination TCP or UDP port specified in the packet. In this case, the host sends the ICMP port unreachable message to the device, which in turn sends the message to the host that sent the packet.

**protocol**

Sends the ICMP unreachable message when TCP or UDP on the destination host is not running. This message is different from the port unreachable message, which indicates that the protocol is running on the host but the requested protocol port is unavailable.

**source-route-fail**

Sends the ICMP unreachable message when the device received a source-routed packet but cannot locate the next hop IP address indicated in the packet Source-Route option.

## Modes

Global configuration mode

# Usage Guidelines

You can disable the device from sending these types of ICMP messages on an individual basis.

**NOTE**
Disabling an ICMP unreachable message type does not change the device ability to forward packets. Disabling ICMP unreachable messages prevents the device from generating or forwarding the unreachable messages.

The **no** form of the command disables the ICMP unreachable messages.

# Examples

The following example enables the ICMP unreachable message when the destination network or subnet of the packet is directly connected to the device, but the host specified in the destination IP address of the packet is not on the network.

```
device(config)# ip icmp unreachable host
```

# ip igmp group-membership-time

Specifies how long an IGMP group remains active on an interface in the absence of a group report.

## Syntax

**ip igmp group-membership-time** *num*

**no ip igmp group-membership-time** *num*

## Command Default

By default, a group will remain active on an interface for 260 seconds in the absence of a group report.

## Parameters

*num*

Number in seconds, from 5 through 26000.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of this command resets the group membership time interval to the default of 260 seconds.

Group membership time defines how long a group will remain active on an interface in the absence of a group report.

## Examples

This example specifies an IGMP (V1 and V2) membership time of 240 seconds.

```
Device(config)# ip igmp group-membership-time 240
```

# ip igmp max-group-address

Configures the maximum number of IGMP group addresses for VRFs.

## Syntax

**ip igmp max-group-address** *num*

**no ip igmp max-group-address** *num*

## Command Default

The default value is 4096.

## Parameters

*num*

Specifies the maximum number of IGMP group addresses available, either for the default VRF or for the specified VRF. The range is 1 through 8192.

## Modes

Global configuration mode

VRF configuration sub-mode

## Usage Guidelines

This command replaces the **system-max igmp-max-group-address** command.

If the **no** form of this command is configured, the maximum number of IGMP group addresses is reset to the default.

## Examples

The following example configures a maximum of 1000 IGMP addresses for the default VRF.

```
device# configure terminal
device(config)# ip igmp max-group-address 1000
```

The following example configures a maximum of 1000 IGMP group addresses for the VRF named vpn1.

```
device# configure terminal
device(config)# vrf vpn1
device(config-vrf-vpn1)# address-family ipv4
device(config-vrf-vpn1-ipv4)# ip igmp max-group-address 1000
```

# ip igmp max-response-time

Defines how long a device waits for an IGMP response from an interface before determining that the group member on that interface is down and removing the interface from the group.

## Syntax

**ip igmp max-response-time** *num*

**no ip igmp max-response-time** *num*

## Command Default

The device waits 10 seconds.

## Parameters

*num*

Number, in seconds, from 1 through 25. The default is 10.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the maximum response time interval to the default of 10 seconds.

## Examples

The following example changes the IGMP (V1 and V2) maximum response time to 8 seconds.

```
device(config)# ip igmp max-response-time 8
```

# ip igmp port-version

Configures an IGMP version recognized by a physical port that is a member of a virtual routing interface.

## Syntax

**ip igmp port-version** *version-number* **ethernet** *unit/slot/port* [ **to ethernet** *unit/slot/port*[ **ethernet** *unit/slot/port...* ] ]

**no ip igmp port-version** *version-number* **ethernet** *unit/slot/port* [ **to ethernet** *unit/slot/port*[ **ethernet** *unit/slot/port..* ] ]

## Command Default

IGMP Version 2 is enabled.

## Parameters

*version-number*
Specifies the version number: 1, 2, or 3. Version 2 is the default.

**ethernet** *unit/slot/port*
Specifies the Ethernet interface.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command restores the default; IGMP Version 2 is enabled.

## Examples

The following example enables IGMP Version 3 on a physical port that is a member of a virtual routing interface. It first enables IGMP Version 2 globally, then enables Version 3 on ports 1/1/3 through 1/1/7 and port 1/2/9. All other ports in this virtual routing interface are configured with IGMP Version 2.

```
device(config)#interface ve 3
device(config-vif-3)# ip igmp version 2
device(config-vif-3)# ip igmp port-version 3 ethernet 1/1/3 to ethernet 1/1/7 ethernet 1/2/9
```

# ip igmp proxy

Configures IGMP proxy on an interface

## Syntax

**ip igmp proxy** [ **group-filter***access-list* ]

**no ip igmp proxy** [ **group-filter***access-list* ]

## Command Default

IGMP proxy is not enabled.

## Parameters

**group-filter**

Specifies filtering out groups in proxy report messages.

*access-list*

Specifies the access list name or number you want filtered out.

## Modes

Interface configuration mode.

## Usage Guidelines

The **no** form of this command disables IGMP proxy on an interface.

IGMP proxy is supported only in PIM dense environments where there are IGMP clients connected to the device. PIM DM must be enabled in passive mode.

IGMP proxy is not supported on interfaces on which PIM sparse mode (SM) or Source Specific Multicast (SSM) is enabled.

Enter the **ip igmp proxy** command without the **group-filter** keyword to remove the group-filter association without disabling the proxy.

## Examples

This example enables IGMP proxy on an interface. It first shows how to configure PIM globally, configure an IP address that will serve as the IGMP proxy for an upstream device on interface 1/3/3, enable PIM passive on the interface, and then enable IGMP proxy.

```
device(config)# router pim
device(config)# interface ethernet 1/3/3
device(config-if-e1000-1/3/3)# ip address 10.95.5.1/24
device(config-if-e1000-1/3/3)# ip pim passive
device(config-if-e1000-1/3/3)# ip igmp proxy
```

The following example filters out the ACL1 group in proxy report messages.

```
device(config)# router pim
device(config)# interface ethernet 1/3/3
device(config-if-e1000-1/3/3)# ip address 10.95.5.1/24
device(config-if-e1000-1/3/3)# ip pim passive
device(config-if-e1000-1/3/3)# ip igmp proxy group-filter ACL1
```

# ip igmp query-interval

Defines how often a device queries an interface for IGMP group membership.

## Syntax

**ip igmp query-interval** *num*

**no ip igmp query-interval** *num*

## Command Default

The query interval is 125 seconds

## Parameters

*num*

Number in seconds, from 2 through 3600. The default is 125.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of this command resets the query interval to the default of 125 seconds.

You must specify a query-interval value that is a little more than twice the group membership time. You can configure the `ip igmp group-membership-time` command to specify the IGMP group membership time.

## Examples

This example sets the IGMP query interval to 120 seconds.

```
Device(config)# ip igmp query-interval 120
```

# ip igmp ssm-map

Enables IGMPv2 SSM mapping and defines the SSM maps between IGMPv2 Group addresses and multicast source addresses.

## Syntax

**ip igmp ssm-map** { *access-list ip-address* | **enable** }

**no ip igmp ssm-map** { *access-list ip-address* | **enable** }

## Command Default

SSM mapping is disabled.

## Parameters

*access-list*

Specifies the name or number of the access list that contains the group multicast address.

*ip-address*

Specifies the source IP address to map to the group multicast address specified in the ACL..

**enable**

Enables IGMPv2 SSM mapping.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command with the **enable** keyword disables IGMPv2 mapping.

The **no** form of the command with the *access-list ip-address* parameters removes an SSM map between an IGMPv2 group address and a multicast source address.

## Examples

The following example enables IGMPv2 mapping.

```
device# configure terminal
device(config)# ip igmp ssm-map enable
```

The following example configures an SSM map between an IGMPv2 group address and a multicast source address.

```
device# configure terminal
device(config)# ip igmp ssm-map 20 10.1.1.1
```

# ip igmp static-group

Configures one or more physical ports to be a permanent (static) member of an IGMP group based on the range or count. Manually adds a port to a multicast group.

## Syntax

**ip igmp static-group** *ip-addr* [ **count** *count-number* | **to** *ip-addr* ] [ **ethernet** *unit**/**slot**/**port* ]

**no ip igmp static-group** *ip-addr* [ **count** *count-number* | **to** *ip-addr* ] [ **ethernet** *unit**/**slot**/**port* ]

## Command Default

The port is not added to multicast group.

## Parameters

*ip-addr*

The address of the static IGMP group.

**count** *count-number*

Specifies the number of continous static groups The range is from 2 through 256.

**ethernet** *unit**/**slot**/**port*

Specifies the ID of the physical port of the VLAN that will be a member of the group.

**to**

Specifies a range of interface.

## Modes

Interface configuration mode

## Usage Guidelines

You can manually add a multicast group to individual ports only. If the port is a member of a virtual routing and forwarding (VRF) interface, you must add the ports to the group individually.

IGMP Version 3 does not support static IGMP group members.

Static IGMP groups are supported only in Layer 3 mode.

The **no** form of this command removes the port from the static group.

## Examples

The following example manually adds port 1/1/2 to multicast group 224.2.2.2.

```
device(config)# interface ethernet 1/1/2
device(config-if-e10000-1/1/2)# ip igmp static-group 224.2.2.2
```

The following example adds port 5/2 that is a member of a VRF interface 1 to multicast group 224.2.2.2.

```
device(config)# interface ve 1
device(config-vif-1)# ip igmp static-group 224.2.2.2 ethernet 5/2/2
```

The following example configures two static groups on virtual ports starting from 226.0.0.1, using the **count** keyword.

```
device(config)# interface ethernet 1/5/1
device(config-if-e1000-1/5/1)# ip igmp static-group 226.0.0.1 count 2 ethernet 1/5/1
```

The following example configures two static groups on virtual ports starting from 226.0.0.1, using the **to** keyword.

```
device(config)# interface ve 10
device(config-vif-10)# ip igmp static-group 226.0.0.1 to 226.0.0.2 ethernet 1/5/1
```

The following example configures two static groups starting from 226.0.0.1, using the **count** keyword.

```
device(config)# interface ethernet 1/5/1
device(config-if-e1000-1/5/1)# ip igmp static-group 226.0.0.1 count 2
```

# ip igmp tracking

Enables tracking and fast leave on an interface.

## Syntax

**ip igmp tracking**

**no ip igmp tracking**

## Command Default

Tracking and fast leave are disabled.

## Modes

Interface configuration mode.

## Usage Guidelines

The **no** form of this command restores the default; tracking and fast leave are disabled.

The IGMP Version 3 fast leave feature is supported in include mode but does not work in exclude mode.

## Examples

This example enables tracking and fast leave on a virtual routing interface.

```
Device(config)# interface ve 13
Device(config-vif-13)# ip igmp tracking
```

This example enables tracking and fast leave on a physical interface.

```
Device(config)# i(config)#interface ethernet 1/2/2
Device(config-if-e10000-1/2/2)# ip igmp tracking
```

# ip igmp version

Specifies the IGMP version on a device.

## Syntax

**ip igmp version** *version-number*

**no ip igmp version** *version-number*

## Command Default

IGMP Version 2 is enabled.

## Parameters

*version-number*

Specifies the version number: 1, 2, or 3. Version 2 is the default.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of this command restores the default; IGMP Version 2 is enabled.

Configure the **ip igmp port-version** command to configure an IGMP version recognized by a physical port that is a member of a virtual routing interface.

## Examples

The following example enables IGMP Version 3 globally.

```
device# configure terminal
device(config)# ip igmp version 3
```

The following example, in interface configuration mode, enables IGMP Version 3 for a physical port.

```
device# configure terminal
device(config)# interface ethernet 1/1/5
device(config-if-1/1/5)# ip igmp version 3
```

The following example, in interface configuration mode, enables IGMP Version 3 for a virtual routing interface on a physical port.

```
device# configure terminal
device(config)# interface ve 3
device(config-vif-1)# ip igmp version 3
```

# ip interface loopback (VXLAN)

Configures the loopback interface for a VXLAN overlay-gateway.

## Syntax

**ip interface loopback** *interface-id*

**no ip interface loopback** *interface-id*

## Parameters

*interface-id*

Specifies the loopback interface ID.

## Modes

Overlay-gateway configuration mode

## Usage Guidelines

The **no** form of the command removes the configured loopback interface as overlay-gateway source interface.

The command is supported only on ICX 7750 devices.

The loopback interface must be configured prior to use in the overlay-gateway configuration.

The loopback interface cannot belong to a user VRF (it must belong to the default VRF).

## Examples

The following example configures loopback interface 1 as the gateway source interface.

```
device# configure terminal
device(config)# overlay-gateway gate1
device(config-overlay-gw-gate1)# type layer2-extension
device(config-overlay-gw-gate1)# map vlan 24 to vni 48
device(config-overlay-gw-gate1)# ip interface loopback 1
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# ip irdp

Enables ICMP Router Discovery Protocol (IRDP) globally.

## Syntax

**ip irdp**

**no ip irdp**

## Command Default

IRDP is disabled.

## Modes

Global configuration mode

## Usage Guidelines

IRDP is used by Layer 3 switches to advertise the IP addresses of its router interfaces to directly attached hosts.

You can enable the feature on a global basis or on an individual port basis. If you enable the feature globally, all ports use the default values for the IRDP parameters. If you leave the feature disabled globally but enable it on individual ports, you also can configure the IRDP parameters on an individual port basis. Use the **ip irdp** command in interface configuration mode to enable IRDP on individual ports.

The **no** form of the command disables IRDP.

## Examples

The following example enables IRDP globally.

```
device(config)# ip idrp
```

# ip irdp (interface)

Enables ICMP Router Discovery Protocol (IRDP) on an interface and configures IRDP parameters.

## Syntax

**ip irdp** { **broadcast** | **multicast** } [ **holdtime** *seconds* ] [ **minadvertinterval** *seconds* ] [ **maxadvertinterval** *seconds* ] [ **preference** *number* ]

**no ip irdp** { **broadcast** | **multicast** } [ **holdtime** *seconds* ] [ **minadvertinterval** *seconds* ] [ **maxadvertinterval** *seconds* ] [ **preference** *number* ]

## Command Default

IRDP is not enabled.

## Parameters

**broadcast**

Configures the Layer 3 switch to send the Router Advertisement as IP broadcasts. This is the default.

**multicast**

Configures the Layer 3 switch to send the Router Advertisement as multicast packets addressed to IPmulticast group 224.0.0.1.

**holdtime** *seconds*

Specifies how long a host that receives a Router Advertisementfrom the Layer 3 switch should consider the advertisement to be valid. The value must be greater than the value of themaxadvertinterval parameter and cannot be greater than 9000. The default is three times the valueof the maxadvertinterval parameter.

**minadvertinterval** *seconds*

Specifies the minimum amount of time the Layer 3 switch can waitbetween sending Router Advertisements. The default is three-fourths (0.75) the value of themaxadvertinterval parameter.

**maxadvertinterval** *seconds*

Specifies the maximum amount of time the Layer 3 switch waits between sending Router Advertisements. You can specify a value from 1 to the current value of theholdtime parameter. The default is 600 seconds.

**preference** *number*

Specifies the IRDP preference level of this Layer 3 switch. If a hostreceives Router Advertisements from multiple routers, the host selects the router interface that sent themessage with the highest interval as the host default gateway. The valid range is from 0 to 4294967296. The default is 0.

## Modes

Interface configuration mode

## Usage Guidelines

IRDP is used by Ruckus Layer 3 switches to advertise the IP addresses of its router interfaces to directly attached hosts.

You can enable the feature on a global basis or on an individual port basis. If you enable the feature globally, all ports use the default values for the IRDP parameters. If you leave the feature disabled globally but enable it on individual ports, you also can configure the IRDP parameters on an individual port basis. You cannot configure IRDP parameters on individual ports if the feature is globally enabled.

The **no** form of the command disables IRDP on the specific interface.

# Examples

The following example enables IRDP on a specific port and changes the maximum advertisement interval for Router Advertisement messages to 400 seconds

```
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# ip irdp maxadvertinterval 400
```

# ip load-sharing

Configures IPv4 load sharing.

## Syntax

**ip load-sharing** [ *number-of-paths* ]

**no ip load-sharing** [ *number-of-paths* ]

## Command Default

Default number of load sharing paths is four.

## Parameters

*number-of-paths*

Specifies the number of paths and can be from 2 through 8, depending on the device you are configuring. On the Ruckus ICX 7750, the value of the num variable can be from 2 through 32.

## Modes

Global configuration mode

## Usage Guidelines

For optimal results, set the maximum number of paths to a value at least as high as the maximum number of equal-cost paths your network typically contains. For example, if the Layer 3 switch you are configuring for IP load sharing has six next-hop routers, set the maximum paths value to six.

The configuration of the maximum number of IP load sharing paths to a value more than 8 is determined by the maximum number of ECMP paths defined at the system level using the **system-max max-ecmp** command. This command is supported only on the Ruckus ICX 7750. You cannot configure the maximum number of IP load sharing paths higher than the value defined at the system level. Also, you cannot configure the maximum number of ECMP paths at the system level to a value less than the configured IP load sharing value.

The **no** form of the command resets the load sharing paths to four.

## Examples

The following example configures IP load sharing paths as 8.

```
device(config)# ip load-sharing 8
```

# ip-mac

Manually configures an IP MAC address on an IP interface.

## Syntax

**ip-mac** *mac-address*

**no ip-mac** *mac-address*

## Command Default

If an IP MAC address is not configured, the IP interface will use the MAC address of the device or the configured stack MAC address.

## Parameters

*MAC-address*

Configures a MAC address on a physical or virtual Ethernet (VE) interface.

## Modes

Interface configuration mode

## Usage Guidelines

Use the **system-max max-ip-mac** command to change the maximum number of MAC addresses to be configured on IP interfaces. The number of MAC addresses to be configured on IP interfaces is a range from 120 to 248 with a default of 120.

Use the **show ip interface** command with a specified interface to view whether a MAC address is configured for the interface.

The **no** form of the command removes the MAC address from the interface.

## Examples

The following example configures a MAC address on Ethernet interface 1/1/6.

```
device# configure terminal
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip-mac aaaa.bbbb.cccc
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ip max-mroute

Configures the maximum number of IPv4 multicast routes that are supported.

## Syntax

**ip max-mroute** *num*

**no ip max-mroute** *num*

## Command Default

No maximum number of supported routes is configured.

## Parameters

*num*

Configures the maximum number of multicast routes supported.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the default (no maximum number of supported routes is configured).

## Examples

The following example configures the maximum number of 20 supported IPv4 multicast routes on the VRF named my_vrf.

```
Device(config)# vrf my_vrf
Device(config)# address-family ipv4
Device(config-vrf)# ip max-mroute 20
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ip mroute

Configures a directly connected static IPv4 multicast route.

## Syntax

**ip mroute** [ **vrf** *vrf-name* ] *ip-address ip-address mask* { **ethernet** *stackid* **/** *slot* **/** *portnum* | **ve** *num* | **tunnel** *num* } [*cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

**no ip mroute** [ **vrf** *vrf-name* ] *ip-address ip-address mask* { **ethernet** *stackid* **/** *slot* **/** *portnum* | **ve** *num* | **tunnel** *num* } [*cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

## Command Default

No static IPv4 multicast route is configured.

## Parameters

**vrf** *vrf-name*

Configures a static mroute for this virtual routing and forwarding (VRF) route.

*ip-address ip-address mask*

Configures the destination IPv4 address and prefix for which the route should be added.

**ethernet** *stackid* **/** *slot* **/***portnum*

Configures an Ethernet interface as the route path.

**ve** *num*

Configures a virtual interface as the route path.

**tunnel** *num*

Configures a tunnel interface as the route path.

*cost*

Configures a metric for comparing the route to other static routes in the static route table that have the same destination. The range is 1-16; the default is 1.

**distance** *distance-value*

Configures the route's administrative distance. The range is 1-255; the default is 1.

**name** *name*

Name for this static route.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form of this command deletes a previously configured directly connected static multicast route.

Connected routes on PIM enabled interfaces are automatically added to the mRTM table.

# Examples

The following example configures a directly connected mroute to network 10.1.1.0/24 on interface ve 10.

```
Device(config-vrf)# ip mroute 10.1.1.0 255.255.255.0 ve 10
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ip mroute (next hop)

Configures a static IPv4 multicast route (mroute) with a next hop..

## Syntax

**ip mroute** [ **vrf** *vrf-name* ] *ip-address ip-address mask next-hop address* [ *cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

**no ip mroute** [ **vrf** *vrf-name* ] *ip-address ip-address mask next-hop address* [ *cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

## Command Default

No next-hop static IPv4 multicast route is configured.

## Parameters

**vrf** *vrf-name*
Configures a static mroute for this virtual routing and forwarding (VRF) route.

*ip-address ip-address mask*
Configures the destination IPv4 address and prefix for which the route should be added.

*next-hop address*
Configures a next-hop address as the route path.

*cost*
Configures a metric for comparing the route to other static routes in the static route table that have the same destination. The range is 1-16; the default is 1.

**distance** *distance-value*
Configures the route's administrative distance. The range is 1 through 255; the default is 1.

**name** *name*
Name for this static route.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form of this command deletes a previously configured next-hop static IPv4 multicast route.

## Examples

The following example configures a next-hop static multicast IPv4 route to network 10.1.1.0/24 with next hop 10.2.1.1.

```
Device(config-vrf)# ip mroute 10.1.1.0 255.255.255.0 10.2.1.1
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ip mroute next-hop-enable-default

Enables the option to use the default multicast route (mroute) to resolve a static IPv4 mroute next hop.

## Syntax

**ip mroute** [ **vrf** *vrf-name* ] **next-hop-enable-default**

**no ip mroute** [ **vrf** *vrf-name* ] **next-hop-enable-default**

## Command Default

Static mroutes are not resolved using the default mroute.

## Parameters

**vrf** *vrf-name*

Configures a static mroute for this virtual routing and forwarding (VRF) route.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form of this command disables the default IPv4 mroute option for next hops.

## Examples

The following example enables the use of the default mroute to resolve a static IPv4 mroute next hop:

```
Device(config-vrf)# ip mroute next-hop-enable-default
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ip mroute next-hop-recursion

Configures the recursion level when using static mroutes to resolve a static mroute next hop.

## Syntax

**ip mroute** [ **vrf** *vrf-name* ] **next-hop-recursion** *num*

**no ip mroute** [ **vrf** *vrf-name* ] **next-hop-recursion**

## Command Default

The recursion level for resolving a static mroute next hop is 3.

## Parameters

**vrf** *vrf-name*

Configures a static mroute for this virtual routing and forwarding (VRF) route.

*num*

Specifies the recursion level used to resolve a static mroute next hop. The range of possible values is from 1 to 10. This is not used in the **no** form.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form restores the default recursion level for resolving a static mroute next hop, which is 3. You do not specify a value for the recursion level.

## Examples

The following example configures the recursion level for resolving a static mroute next hop to 7:

```
device(config)# vrf vrf2
device(config-vrf-vrf2)# ip mroute next-hop-recursion 7
```

The following example configures the recursion level for resolving a static mroute next hop to 2:

```
device(config)# vrf vrf2
device(config-vrf-vrf2)# ip mroute next-hop-recursion 2
```

The following example restores the default recursion level of 3 for resolving a static mroute next hop:

```
device(config)# vrf vrf2
device(config-vrf-vrf2)# no ip mroute next-hop-recursion
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ip mtu

Changes the MTU for a specific interface.

## Syntax

**ip mtu** *value*

**no ip mtu** *value*

## Command Default

1500 bytes for Ethernet II encapsulation and 1492 bytes for SNAP encapsulation. When jumbo mode is enabled, the default is 9216.

## Parameters

*value*

Specifies the MTU. Ethernet II packets can hold IP packets from 576 through 1500 bytes long. If jumbo mode is enabled, Ethernet II packets can hold IP packets up to 10,218 bytes long. Ethernet SNAP packets can hold IP packets from 576 through 1492 bytes long. If jumbo mode is enabled, SNAP packets can hold IP packets up to 10,214 bytes long. The default MTU for Ethernet IIpackets is 1500. The default MTU for SNAP packets is 1492.

## Modes

Interface configuration mode

## Usage Guidelines

If you set the MTU of a port to a value lower than the global MTU and from 576 through 1499, the port fragments the packets. However, if the port MTU is exactly 1500 and this is larger than the global MTU, the port drops the packets. The minimum IPv4 MTU values for both physical and virtual interfaces are 1280.

You must save the configuration change and then reload the software to enable the configuration.

The **no** form of the command resets the default MTU values.

## Examples

The following example configures the IP MTU as 1300.

```
device(config)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip mtu 1300
device(config-if-e1000-1/1/5)# write memory
device(config-if-e1000-1/1/5)# end
device# reload
```

# ip multicast

Configures the IGMP mode on a specific VLAN or on all VLANs on a device as active or passive.

## Syntax

**ip multicast** [ **vlan** | *vlan-id* ] [ **active** | **passive** ]

**no ip multicast**

## Command Default

IGMP mode is passive.

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

**active**

Configures IGMP active mode, that is, the device actively sends out IGMP queries to identify multicast groups on the network and makes entries in the IGMP table based on the group membership reports it receives.

**passive**

Configures IGMP passive mode, that is, the device does not send queries but forwards reports to the router ports that receive queries. When passive mode is configured on a VLAN, queries are forwarded to the entire VLAN.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

The **no** form of this command returns the device to the previous IGMP mode.

When entered without the **vlan** keyword, this command configures active or passive IGMP mode on all VLANs.

Routers in the network generally handle mode. Configure active IGMP mode only on a device is in a standalone Layer 2 Switched network with no external IP multicast router attachments. If you want to configure active IGMP mode on a device in such a network, you should do so on only one device and leave the others configured as passive.

The IGMP mode configured on a VLAN overrides the mode configured globally.

## Examples

The following example globally configures IGMP mode as active.

```
device#configure terminal
device(config)#ip multicast active
```

This example configures IGMP mode as active on VLAN 20.

```
device#configure terminal
device(config)#config vlan 20
device(config-vlan-20)#ip multicast active
```

# ip multicast age-interval

Configures the time that group entries can remain in an IGMP group table on a specific VLAN or on all VLANs.

## Syntax

**ip multicast age-interval** [ **vlan** *vlan-id* ] *interval*

**no ip multicast age-interval** [ **vlan** *vlan-id* ] *interval*

## Command Default

Group entries can remain in the IGMP group table for up to 260 seconds.

## Parameters

**vlan** *vlan-id*

Specifies a VLAN.

*interval*

Specifies time, in seconds, that group entries can remain in the IGMP group table. The range is 20 through 26000 seconds. The default is 260 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default age interval to 260 seconds.

When entered without the **vlan** keyword, this command configures the time that group entries can remain in an IGMP group table on all VLANs.

When a device receives a group membership report it makes an entry for that group in the IGMP group table. You can configure the **ip multicast age-interval** to specify how long the entry can remain in the table before the device receives another group membership report. When multiple devices are connected, they must all be configured for the same age interval, which must be at least twice the length of the query interval, so that missing one report does not stop traffic.

Non-querier age intervals must be the same as the age interval of the querier.

## Examples

This example configures the IGMP group-table age interval to 280 seconds.

```
device#configure terminal
device(config)#ip multicast age-interval 280
```

# ip multicast disable-flooding

Disables the flooding of unregistered IPv4 multicast frames in an IGMP-snooping-enabled VLAN.

## Syntax

**ip multicast disable-flooding**

**no ip multicast disable-flooding**

## Command Default

The device floods unregistered IPv4 multicast frames in an IGMP-snooping-enabled VLAN.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command enables the flooding of unregistered IPv4 multicast frames in an IGMP-snooping-enabled VLAN.

Support for this command on the Ruckus ICX 7750 was introduced in FastIron 8.0.10d. In releases prior to FastIron 8.0.30, support for this command on the Ruckus ICX 7750 was for devices in standalone mode only.

Support for this command on the Ruckus ICX 7450 and Ruckus ICX 7250 was introduced in FastIron 8.0.30.

After the hardware forwarding database (FDB) entry is made, the multicast traffic is switched only to the VLAN hosts that are members of the multicast group. This can avoid congestion and loss of traffic on the ports that have not subscribed to this IPv4 multicast traffic.

## Examples

The following example disables flooding of unregistered IPv4 multicast frames.

```
device(config)# ip multicast disable-flooding
```

## History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced. |

# ip multicast leave-wait-time

Configures the wait time before stopping traffic to a port when a leave message is received.

## Syntax

**ip multicast leave-wait-time** *num*

**no ip multicast leave-wait-time** *num*

## Command Default

The wait time is 2 seconds.

## Parameters

*num*

Specifies the time, in seconds, the device should wait before stopping traffic to a port when a leave message is received The range is 1 through 5 seconds. The default is 2 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default wait time.

The device sends group-specific queries once per second to ask if any client in the same port still needs this group. Because of internal timer granularity, the actual wait time is between n and (n+1) seconds (n is the configured value).

## Examples

This example configures the maximum time a client can wait before responding to a query to 1 second.

```
Device(config)#ip multicast leave-wait-time 1
```

# ip multicast max-response-time

Sets the maximum number of seconds a client (IPv4) can wait before responding to a query sent by the device.

## Syntax

**ip multicast max-response-time** *interval*

**no ip multicast max-response-time** *interval*

## Command Default

The wait time is 10 seconds.

## Parameters

*interval*

Specifies the maximum time, in seconds, a client can wait before responding to a query sent by the switch. The range is 1 through 25 seconds. The default is 10 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum interval.

## Examples

This example configures the maximum time a client can wait before responding to a query to 5 seconds.

```
device(config)# ip multicast max-response-time 5
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was modified to increase the range of the maximum response time from 1 through 10 seconds to 1 through 25 seconds. |

# ip multicast mcache-age

Configures the time for an mcache to age out when it does not receive traffic.

## Syntax

**ip multicast mcache-age** *num*

**no ip multicast mcache-age**

## Command Default

The mcache ages out after the default age-out interval, which is 180 seconds for ICX 7750, ICX 7650, ICX 7450, and ICX 7250 devices.

## Parameters

*num*

Specifies the time, in multiples of 60 seconds, the device should wait before stopping traffic to a port when a leave message is received The range is 60 through 3600 seconds, in multiples of 60.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default mcache age-out time.

Multicast traffic is hardware switched. One minute before aging out an mcache, the device mirrors a packet of this mcache to CPU to reset the age. If no data traffic arrives within 60 seconds, this mcache is deleted. Configuring a lower age-out time removes resources consumed by idle streams quickly, but it mirrors packets to CPU often. Configure a higher value only when data streams are arriving consistently.

**NOTE**
Multicast mcache may not expire according to the configured time. You may notice a delay of 0 to 60 seconds over the configured value.

## Examples

This example configures the time for an mcache to age out to 180 seconds.

```
device(config)# ip multicast mcache-age 180
```

## History

| Release version | Command history |
|---|---|
| 08.0.60 | Added note about multicast mcache expiry. |

# ip multicast optimization

Enables or disables IP multicast (IPMC) entry optimization for Layer 2 IPv4 multicast flows.

## Syntax

**ip multicast optimization oif-list all**

**no ip multicast optimization oif-list all**

## Command Default

IPMC entry optimization is disabled by default on ICX 7750 devices, and enabled by default on ICX 7450 and ICX 7250 devices.

## Parameters

**oif-list**

Shares the Output Interface Lists across entries.

**all**

Specifies all types of Output Interface Lists.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables hardware entry optimization for Layer 2 IPv4 multicast flows. The command must be followed by the **write memory** command and the **reload** command for the changes to take effect.

## Examples

The following example enables hardware entry optimization for Layer 2 IPv4 multicast flows.

```
device(config)# ip multicast optimization oif-list all
device(config)# write memory
device(config)# exit
device# reload
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ip multicast query-interval

Configures how often the device sends general queries when IP multicast traffic reduction is set to active mode.

## Syntax

**ip multicast query-interval** *interval*

**no ip multicast query-interval** *interval*

## Command Default

The query interval is 125 seconds.

## Parameters

*interval*

Specifies the time, in seconds, between queries. The range is 10 through 3600 seconds. The default is 125 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the query interval to 125 seconds.

You can configure this command only when IP multicast traffic reduction is set to active IGMP snooping mode.

When multiple queries are connected, they must all be configured for the same interval.

## Examples

This example configures the time between queries to 120 seconds.

```
Device(config)#ip multicast query-interval 120
```

# ip multicast report-control

Limits report forwarding within the same multicast group to no more than once every 10 seconds.

## Syntax

**ip multicast report-control**

**no ip multicast report-control**

## Command Default

A device in passive mode forwards reports and leave messages from clients to the upstream router ports that are receiving queries.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default.

> **NOTE**
> This feature applies to IGMP V2 only. The leave messages are not rate limited.

This rate-limiting does not apply to the first report answering a group-specific query.

Configure this command to alleviate report storms from many clients answering the upstream router query.

The **ip multicast report-control** command was formerly named **ip igmp-report-control** . You can still configure the command as **ip igmp-report-control** ; however, it is renamed when you configure the **show configuration** command.

## Examples

This example limits the rate of report forwarding within the same multicast group.

```
Device(config)#ip multicast report-control
```

# ip multicast verbose-off

Turns off the error or warning messages displayed by the device when it runs out of software resources or when it receives packets with the wrong checksum or groups.

## Syntax

**ip multicast verbose-off**

**no ip multicast verbose-off**

## Command Default

Error and warning messages are displayed.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores display of error and warning messages .

Error and warning messages are rate-limited.

## Examples

This example turns off error or warning messages .

```
Device(config)#ip multicast verbose-off
```

# ip multicast version

Configures the IGMP version for snooping globally.

## Syntax

**ip multicast version** [ **2** | **3** ]

**no ip multicast version**

## Command Default

IGMP version 2 is configured.

## Parameters

**2**

Configures IGMP version 2.

**3**

Configures IGMP version 3.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the version to IGMP version 2.

If Layer 3 multicast routing is enabled on the device, Layer 2 IGMP snooping is automatically enabled.

See the description of the **multicast version** command for information on how to configure the IGMP version on a VLAN.

See the description of the **multicast port-version** command for information on how to configure the IGMP version on an individual port

## Examples

This example specifies IGMP version 3 on a device.

```
Device(config)#ip multicast version 3
```

# ip multicast-boundary

Defines boundaries for PIM enabled interfaces.

## Syntax

**ip multicast-boundary** *acl-spec*

**no ip multicast-boundary** *acl-spec*

## Command Default

Boundaries are not defined.

## Parameters

*acl-spec*

Specifies the number or name identifying an access list that controls the range of group addresses affected by the boundary.

## Modes

VE interface configuration mode

## Usage Guidelines

The **no** form of this command removes the boundary on a PIM enabled interface.

You can use standard ACL syntax to configure an access list.

## Examples

The following example defines a boundary named MyAccessList for a PIM enabled interface.

```
device(config)# ip multicast-boundary MyAccessList
```

# ip multicast-debug-mode

Enables global multicast debug mode for all VRFs.

## Syntax

**ip multicast-debug-mode**

**no ip multicast-routing**

## Command Default

Support for multicast debug mode is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default multicast debug mode.

## Examples

This example shows how to enable global support for multicast debug mode.

```
Device(config)#ip multicast-debug-mode
```

# ip multicast-nonstop-routing

Globally enables multicast non-stop routing for all virtual routing and forwarding (VRF) instances.

## Syntax

**ip multicast-nonstop-routing**

**no ip multicast-nonstop-routing**

## Command Default

Multicast non-stop routing is not enabled on VRFs.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default non-stop routing.

## Examples

The following example globally enables multicast non-stop routing for all VRFs.

```
device#configure terminal
device(config)#ip multicast-nonstop-routing
```

# ip multicast-routing optimization

Enables or disables IP multicast (IPMC) hardware entry optimization for Layer 3 IPv4 multicast flows.

## Syntax

**ip multicast-routing optimization oif-list all**

**no ip multicast-routing optimization oif-list all**

## Command Default

Hardware entry optimization is disabled by default on ICX 7750 devices and enabled by default on ICX 7450 and 7250 devices.

## Parameters

**oif-list**

Shares the Output Interface Lists across entries.

**all**

Specifies all types of Output Interface Lists.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables optimization for IPv4 multicast flows. Multicast routing entries are deleted and recreated when optimization is enabled or disabled on all VRFs. The command must be followed by the **write memory** command and the **reload** command for the changes to take effect.

## Examples

The following example enables IPMC hardware entry optimization for IPv4 multicast flows.

```
device(config)# ip multicast-routing optimization oif-list all
device(config)# write memory
device(config)# exit
device# reload
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ip multicast-routing rpf-check mac-movement

Triggers Reverse Path Forwarding (RPF) check on MAC movement for directly connected sources and sends a MAC address movement notification to the Protocol Independent Multicast (PIM) module which results in PIM convergence.

## Syntax

**ip multicast-routing rpf-check mac-movement**

**no ip multicast-routing rpf-check mac-movement**

## Command Default

RPF check on MAC movement for directly connected sources is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

PIM convergence on MAC movement is applicable only in a topology where the multicast source port and PIM routers are in the same Layer 2 domain.

The **ip multicast-routing rpf-check mac-movement** command is not supported on the Ruckus ICX 7250 devices.

The **no** form of the command disables RPF check on MAC movement for directly connected sources.

## Examples

The following example configures RPF check on MAC movement for directly connected sources.

```
device(config)# ip multicast-routing rpf-check mac-movement
```

## History

| Release version | Command history |
|---|---|
| 08.0.10h | This command was introduced. |
| 08.0.30 | Support for the **ip multicast-routing rpf-check mac-movement** command was added in 08.0.30 and later releases. |

# ip ospf active

Sets a specific OSPF interface to active.

## Syntax

**ip ospf active**

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **ip ospf active** command on each interface participating in adjacency formation. This command overrides the global passive setting on that interface, and enables transmission of OSPF control packets.

## Examples

The following example sets a specific OSPFv2 virtual Ethernet (VE) interface to active.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ip ospf active
```

# ip ospf area

Enables OSPFv2 on an interface.

## Syntax

**ip ospf area** *area-id* | *ip-addr*

**no ip ospf area**

## Command Default

Disabled.

## Parameters

*area-id*

Area ID in decimal format. Valid values range from 1 through 2147483647.

*ip-addr*

Area ID in IP address format.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command disables OSPFv2 on the interface.

## Examples

The following example enables a configured OSPFv2 area named 0 on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ip ospf area 0
```

# ip ospf authentication

Configures MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication for Open Shortest Path First version 2 (OSPFv2).

## Syntax

**ip ospf authentication** { **md5** | **hmac-sha-1** | **hmac-sha-256**} **key-id** *key-id-val* **key** *key-string*

**no ip ospf authentication** { **md5** | **hmac-sha-1** | **hmac-sha-256**} *key-id-val* **key** *key-string*

## Command Default

MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication is disabled by default.

## Parameters

**md5**
Specifies MD5 authentication.

**HMAC-SHA-1**
Specifies HMAC-SHA-1 authentication.

**HMAC-SHA-256**
Specifies HMAC-SHA-256 authentication.

**key-id** *key-id-val*
Identifies the number of the MD5, HMAC-SHA-1 or HMAC-SHA-256 algorithm. The number can be from 1 through 255.

**key** *key-string*
Sets the corresponding key string to be used with the MD5, HMAC-SHA-1 or HMAC-SHA-256 algorithm. The recommended key string length is 1 through 63 characters.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication configuration on the OSPFv2 interface to which you are connected.

The **no** form of the command removes the MD5, HMAC-SHA-1 or HMAC-SHA-256 authentication configuration from the OSPFv2 interface.

# Examples

The following example sets HMAC-SHA-1 authentication with key ID '10' and the password key "mypasswordkey", on the OSPFv2 interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ip ospf authentication hmac-sha-1 key-id 10 key mypasswordkey
```

# History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ip ospf authentication key-activation-wait-time

Configures the time before an authentication key change is activated for an Open Shortest Path First version 2 (OSPFv2) interface.

## Syntax

**ip ospf authentication key-activation-wait-time** *wait-time*

**no ip ospf authentication key-activation-wait-time** *wait-time*

## Parameters

*wait-time*

Specifies the time before an authentication key change takes place. The wait time can be set from 0 through 14400 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the wait time before an authentication key change takes place on the interface to which you are connected.

The **no** form of the command resets the wait time to the default of 300 seconds.

## Examples

The following example sets the wait time before an authentication key change to 600 seconds on the OSPFv2 interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ip ospf authentication key-activation-wait-time 600
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ip ospf authentication keychain

Configures Open Shortest Path First version 2 (OSPFv2) authentication using the keychain authentication module.

## Syntax

**ip ospf authentication keychain** *keychain-name*

**no ip ospf authentication keychain** *keychain-name*

## Parameters

*keychain-name*

Specifies the name of the keychain that OSPFv2 uses to authenticate the packets.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The keychain authentication module provides the OSPFv2 protocol the option to automatically change the key ID and cryptographic algorithm without manual intervention.

With this configuration, OSPFv2 requests the keychain authentication module for all active keys in the keychain and selects the keys for sending and accepting the packets.

The **no** form of the command removes keychain authentication from the OSPFv2 interface configuration.

## Examples

The following example configures OSPFv2 to use the keychain authentication module with the "xtreme" keychain.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-11)# ip ospf authentication keychain xtreme
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ip ospf authentication plain-text

Configures simple password-based authentication for Open Shortest Path First version 2 (OSPFv2).

## Syntax

**ip ospf authentication plain-text** *key-string*

**no ip ospf authentication plain-text** *key-string*

## Command Default

Password-based authentication is disabled by default.

## Parameters

*key-string*

Sets the authentication password. The key string is unencrypted and appended to the outgoing message.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset simple password-based authentication on the OSPFv2 interface to which you are connected.

The **no** form of the command removes plain text authentication from the OSPFv2 interface configuration.

## Examples

The following example configures the authentication password "mystring" in plain text on the OSPFv2 interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-11)# ip ospf authentication plain-text mystring
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ip ospf cost

Configures cost for a specific interface.

## Syntax

**ip ospf cost** *value*

**no ip ospf cost**

## Command Default

Cost value is 1.

## Parameters

*value*

Cost value. Valid values range from 1 through 65535. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the OSPFv2 cost on the interface. If the cost is not configured with this command, OSPFv2 calculates the value from the reference and interface bandwidths.

You can modify the cost to differentiate between 100 Mbps, 1 Gbps, and 10 Gbps. The default cost is calculated by dividing 100 million by the bandwidth. For 10 Mbps links, the cost is 10. The cost for 100 Mbps, 1 Gbps, and 10 Gbps links is 1, because the speed of 100 Mbps and 10 Gbps was not in use at the time the OSPF cost formula was devised.

The **no** form of the command disables the configured cost.

## Examples

The following example sets the cost to 600 on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-11)# ospf cost 600
```

# ip ospf database-filter

Configures filters for different types of outgoing Link State Advertisements (LSAs).

## Syntax

**ip ospf database-filter all out**

**ip ospf database-filter all-external** { **allow-default out** | **allow-default-and-type-4 out** | **out** }

**ip ospf database-filter all-summary-external** { **allow-default out** | **allow-default-and-type-4 out** | **out** }

**no ip ospf database-filter all out**

**no ip ospf database-filter all-external**

**no ip ospf database-filter all-summary-external**

## Command Default

All filters are disabled.

## Parameters

**all out**
Blocks all LSAs.

**all-external**
Blocks all external LSAs.

**allow-default-and-type-4**
Allows default-route LSAs and Type 4 LSAs, but block all other LSAs.

**allow-default-out**
Allows default-route LSAs, but block all other LSAs.

**out**
Filters outgoing LSAs.

**all-summary-external**
Blocks all summary (Type 3) and external (type 5) LSAs.

## Modes

Interface subtype configuration mode

## Usage Guidelines

By default, the device floods all outbound LSAs on all the OSPFv2 interfaces within an area. You can configure a filter to block outbound LSAs on an OSPF interface. This feature is particularly useful when you want to block LSAs from some, but not all, of the interfaces attached to the area. When enabled, this command blocks the specified outgoing LSAs on the interface. Some cases where you might want to enable filters are:

- To control the information being advertised to the network.

- To use a passive router for debugging only.

Enter **no ip ospf database-filter** followed by the appropriate operands to disable this configuration.

**NOTE**
You cannot block LSAs on virtual links and LSA filtering is not supported on sham links.

# Examples

The following example applies a filter to block flooding of all LSAs on a specific OSPF Ethernet interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip ospf database-filter all-out
```

# ip ospf dead-interval

Configures the neighbor dead interval, which is the number of seconds that a neighbor router waits for a hello packet from the device before declaring the router down.

## Syntax

**ip ospf dead-interval** *interval*

**no ip ospf dead-interval**

## Command Default

The specified time period is 40 seconds.

## Parameters

*interval*

Dead interval in seconds. Valid values range from 3 through 65535 seconds. The default is 40.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the dead interval, the hello interval is automatically changed to a value that is one fourth that of the new dead interval, unless the hello interval is also explicitly configured using the **ip ospf hello-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is ¼ times that of the dead interval.
- If the OSPF hello interval and dead interval are set to more aggressive levels than 1:4 seconds respectively, the OSPF protocol might flap when the **write memory** command is used or in the case of any high CPU.

The **running-config** command displays only explicitly configured values of the hello interval, which means that a value that was automatically changed as the result of a dead-interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the dead interval to 200 on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ip ospf dead-interval 200
```

# ip ospf hello-interval

Configures the hello interval, which is the length of time between the transmission of hello packets that this interface sends to neighbor routers.

## Syntax

**ip ospf hello-interval** *interval*

**no ip ospf hello-interval**

## Command Default

The default value is 10 seconds.

## Parameters

*interval*

Hello interval in seconds. Valid values range from 1 through 65535.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the hello interval, the dead interval is automatically changed to a value that is four times that of the new hello interval, unless the dead interval is also explicitly configured using the **ip ospf dead-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is ¼ times that of the dead interval.
- If the OSPF hello interval and dead interval are set to more aggressive levels than 1:4 seconds respectively, the OSPF protocol might flap when the **write memory** command is used or in the case of any high CPU.

The **running-config** command displays only explicitly configured values of the dead interval, which means that a value that was automatically changed as the result of a hello-interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the hello interval to 50 on a specific OSPFv2 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ip ospf hello-interval 50
```

# ip ospf mtu-ignore

Enables or disables maximum transmission unit (MTU) match checking.

## Syntax

**ip ospf mtu-ignore**

**no ip ospf mtu-ignore**

## Command Default

Enabled

## Modes

Interface subtype configuration mode

## Usage Guidelines

In default operation, the IP MTU on both sides of an OSPFv2 link must be the same, and a check of the MTU is performed when Hello packets are first exchanged.

The **no** form of the command re-enables MTU-match checking on a specific interface if it has been disabled.

## Examples

The following example disables MTU-match checking on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ip ospf mtu-ignore
```

The following example re-enables MTU-match checking on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# no ip ospf mtu-ignore
```

# ip ospf network

Configures the network type for the interface. Point-to-point can support unnumbered links, which requires less processing by OSPF.

## Syntax

**ip ospf network** { **broadcast** | **non-broadcast** | **point-to-point** }

**no ip ospf network**

## Command Default

Network type is broadcast.

## Parameters

**broadcast**
Network type is broadcast.

**non-broadcast**
Network type is non-broadcast. An interface can be configured to send OSPF traffic to its neighbor as unicast packets rather than multicast packets.

**point-to-point**
Network type is point-to-point.

## Modes

Interface subtype configuration mode

## Usage Guidelines

On a non-broadcast interface, the devices at either end of the interface must configure non-broadcast interface type and the neighbor IP address. There is no restriction on the number of devices sharing a non-broadcast interface.

To configure an OSPF interface as a non-broadcast interface, the feature must be enabled on a physical interface or a VE, following the **ip ospf area** statement, and then specify the IP address of the neighbor in the OSPF configuration. The non-broadcast interface configuration must be done on the OSPF devices at either end of the link.

The **no** form of the command removes the network-type configuration.

## Examples

The following example configures an OSPFv2 point-to-point link on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ip ospf network point-to-point
```

The following example configures an OSPFv2 broadcast link on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ip ospf network broadcast
```

# ip ospf passive

Sets a specific OSPFv2interface to passive.

## Syntax

**ip ospf passive**

**no ip ospf passive**

## Command Default

All OSPF interfaces are active.

## Modes

Interface subtype configuration mode

## Usage Guidelines

When you configure an OSPF interface to be passive, that interface does not send or receive OSPF route updates. Since a passive interface does not send or receive route information, the interface is in effect a stub network.

You might want to set an interface to passive mode if:

- You are planning to use the router mostly for debugging purposes.
- The router is a stub and does not route traffic.

The **no** form of the command sets an interface back to active.

## Examples

The following example sets a specific OSPFv2 virtual Ethernet (VE) interface to passive.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ip ospf passive
```

# ip ospf priority

Configures priority for designated router (DR) election.

## Syntax

**ip ospf priority** *value*

**no ip ospf priority**

## Command Default

The default value is 1.

## Parameters

*value*

Priority value. Valid values range from 0 through 255.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The OSPFv2 router assigned the highest priority becomes the designated router, and the OSPFv2 router with the second-highest priority becomes the backup router.

If you set the priority to 0, the device does not participate in DR and BDR election.

The **no** form of the command restores the default value.

## Examples

The following example sets a priority of 10 for the OSPFv2 router that is connected to an OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf priority 10
```

# ip ospf retransmit-interval

Configures the retransmit interval. The retransmit interval is the time between Link-State Advertisement (LSA) retransmissions to adjacent routers for a given interface.

## Syntax

**ip ospf retransmit-interval** *interval*

**no ip ospf retransmit-interval**

## Command Default

The interval is 5 seconds.

## Parameters

*interval*

Retransmit interval in seconds. Valid values range from 0 through 3600 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command resets the retransmit interval to its default.

## Examples

The following example sets the retransmit interval to 8 for all OSPFv2 devices on an OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ip ospf retransmit-interval 8
```

# ip ospf transmit-delay

Configures transmit delay for link-update packets. The transmit delay is the estimated time required for OSPFv2 to send link-state update packets on the interface to which you are connected.

## Syntax

**ip ospf transmit-delay** *value*

**no ip ospf transmit-delay**

## Command Default

The transmit delay is set to 1 second.

## Parameters

*value*

Transmit delay in seconds. Valid values range from 0 through 3600 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

## Examples

The following example sets a transmit delay of 25 seconds for devices on a specific OSPFv2 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ip ospf transmit-delay 25
```

# ip pcp-remark

Enables remarking of the priority code point (PCP) field in the VLAN header for all received tagged packets.

## Syntax

**ip pcp-remark** *pcp-value*

**no ip pcp-remark** *pcp-value*

## Command Default

PCP remarking is disabled.

## Parameters

*pcp-value*

Specifies the PCP value ranges you are remarking.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of this command disables PCP remarking.

In Interface configuration mode, the command enables PCP remarking for each port. The command can be configured only on Layer 2 ports. The configuration can be done on a physical port, LAG, and VE port.

## Examples

The following example globally enables remarking of received tagged packets when the PCP bit value is 4.

```
Device(config)# ip pcp-remark 4
```

The following example enables remarking of received tagged packets on a specific port when the PCP bit value is 5.

```
Device(config)# interface ethernet1/1/1
Device(config-if-e1000-1/1/1)# ip pcp-remark 5
```

# ip pim

Configures PIM in Dense mode on an interface.

## Syntax

**ip pim** [ **passive** ]

**no ip pim** [ **passive** ]

## Command Default

PIM is not enabled.

## Parameters

**passive**

Specifies PIM passive mode on the interface.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command disables PIM.

You must enable PIM globally before you enable it on an interface.

You must enable PIM on an interface before you can configure PIM passive on it.

Support for the **ip pim passive** command is implemented at Layer 3 interface (Ethernet or virtual Ethernet) level.

Because the loopback interfaces are never used to form PIM neighbors, the **ip pim passive** command is not supported on loopback interfaces.

The sent and received statistics of a PIM Hello message are not changed for an interface while it is configured as PIM passive.

## Examples

This example enables PIM globally, then enables it on interface 3.

```
Device(config)# router pim
Device(config-pim-router)# interface ethernet 1/1/3
Device(config-if-e10000-1/1/3)# ip address 207.95.5.1/24
Device(config-if-e10000-1/1/3)# ip pim
```

This example enables PIM passive on an interface.

```
Device(config)# router pim
device(config-pim-router)#exit
Device(config)#interface ethernet 2
Device(config-if-e1000-2)#ip pim
Device(config-if-e1000-2)#ip pim passive
Device(config-if-e1000-2)#exit
Device(config)#interface ve 2
Device(config-vif-2)#ip pim-sparse
Device(config-vif-2)#ip pim passive
Device(config-vif-2)#exit
```

# ip pim border

Configures PIM parameters on an interface on a PIM Sparse border.

## Syntax

**ip pim border**

**no ip pim border**

## Command Default

The interface is not configured as a border device.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command removes the boundary on a PIM-enabled interface.

You can configure this command only in a PIM Sparse domain, that is, you must configure the **ip pim-sparse** command before you configure the **ip pim border** command.

## Examples

This example adds an IPv4 interface to port 1/2/2, enables PIM Sparse on the interface and configures it as a border device.

```
Device(config)# interface ethernet 1/2/2
Device(config-if-e10000-1/2/2)# ip address 207.95.7.1 255.255.255.0
Device(config-if-e10000-1/2/2)# ip pim-sparse
Device(config-if-e10000-1/2/2)# ip pim border
```

# ip pim dr-priority

Configures the designated router (DR) priority on IPv4 interfaces.

## Syntax

**ip pim dr-priority** *priority-value*

**no ip pim dr-priority** *priority-value*

## Command Default

The default DR priority value is 1.

## Parameters

*priority-value*

Specifies the DR priority value as an integer. The range is 0 through 65535.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command restores the default DR priority value, 1.

You must enable PIM globally before you enable it on an interface.

You can configure the **ip pim dr-priority** command in either Dense mode (DM) or Sparse mode (SM).

If more than one device has the same DR priority on a subnet (as in the case of default DR priority on all), the device with the numerically highest IP address on that subnet is elected as the DR.

The DR priority information is used in the DR election only if all the PIM devices connected to the subnet support the DR priority option. If at least one PIM device on the subnet does not support this option, the DR election falls back to the backwards compatibility mode in which the device with the numerically highest IP address on the subnet is declared the DR regardless of the DR priority values.

## Examples

This example configures a DR priority value of 50.

```
device(config)# interface ethernet 1/3/24
device(config-if-e10000-1/3/24)# ip pim dr-priority 50
```

This example configures a DR priority value of 50.

```
device(config)# interface ethernet 1/3/24
device(config-if-e10000-1/3/24)# ip pim dr-priority 50
```

# ip pim neighbor-filter

Determines which devices can become PIM neighbors.

## Syntax

**ip pim neighbor-filter** { *acl-name* | *acl-id* }

**no ip pim neighbor-filter** { *acl-name* | *acl-id* }

## Command Default

Neighbor filtering is not applied on the interface.

## Parameters

*acl-name*

Specifies an ACL as an ASCII string.

*acl-id*

Specifies either a standard ACL as a number in the range 1 to 99 or an extended ACL as a number in the range 100 to 199.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command removes any neighbor filtering applied on the interface.

You must enable PIM globally before you enable it on an interface.

You can configure the **ip pim neighbor-filter** command in either Dense mode (DM) or Sparse mode (SM).

Configure the **access-list** command to create an access-control list (ACL)that specifies the devices you want to permit and deny participation in PIM

## Examples

This example prevents the host from becoming a PIM neighbor on interface Ethernet 1/3/24.

```
Device(config)# interface ethernet 1/3/24
Device(config-if-e10000-1/3/24)# ip pim neighbor-filter
```

This example configures an ACL named 10 to deny a host and then prevents that host, 10.10.10.2, identified in that ACL from becoming a PIM neighbor on interface Ethernet 1/3/24.

```
Device(config)# access-list  10  deny  host 10.10.10.2
Device(config)# access-list  10  permit any
Device(config)# interface ethernet 1/3/24
Device(config-if-e10000-1/3/24)# ip pim neighbor-filter 10
```

# History

| Release version | Command history |
|---|---|
| 8.0.20a | This command was introduced. |

# ip pim-sparse

Enables PIM Sparse on an interface that is connected to the PIM Sparse network.

## Syntax

**ip pim-sparse** [ **passive** ]

**no ip pim-sparse** [ **passive** ]

## Command Default

PIM Sparse is not enabled on the interface.

## Parameters

**passive**

Specifies PIM passive mode on the interface.

## Modes

Interface configuration mode

## Usage Guidelines

You must enable PIM Sparse globally before you enable it on an interface.

If the interface is on the border of the PIM Sparse domain, you also must configure the **ip pim border** command.

The **no ip pim-sparse** command disables PIM Sparse.

The **no ip pim-sparse passive** command disables PIM passive mode on the interface.

## Examples

The following example adds an IP interface to port 1/2/2, then enable PIM Sparse on the interface.

```
device# configure terminal
device(config)# interface ethernet 1/2/2
device(config-if-e10000-1/2/2)# ip address 207.95.7.1 255.255.255.0
device(config-if-e10000-1/2/2)# ip pim-sparse
```

# ip pimsm-snooping

Enables PIM Sparse mode (SM) traffic snooping globally.

## Syntax

**ip pimsm-snooping**

**no ip pimsm-snooping**

## Command Default

PIM SM traffic snooping is disabled.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

The **no** form of this command disables PIM SM traffic snooping.

The device must be in passive mode before it can be configured for PIM SM snooping.

Use PIM SM snooping only in topologies where multiple PIM sparse routers connect through a device. PIM SM snooping does not work on a PIM dense mode router that does not send join messages and on which traffic to PIM dense ports is stopped. A PIM SM snooping-enabled device displays a warning if it receives PIM dense join or prune messages.

When PIM SM snooping is enabled globally, you can override the global setting and disable it for a specific VLAN.

## Examples

This example shows how to enable PIM SM traffic snooping.

```
Device(config)# ip pimsm-snooping
```

This example overrides the global setting and disable PIM SM traffic snooping on VLAN 20.

```
Device(config)# vlan 20
Device(config-vlan-20)# no ip pimsm-snooping
```

# ip policy route-map

Enables policy-based routing (PBR).

## Syntax

**ip policy route-map** *map-name*

**no ip policy route-map** *map-name*

## Command Default

PBR is not enabled.

## Parameters

*map-name*

Specifies the name of the route map.

## Modes

Global configuration mode

Interface configuration mode

Virtual interface configuration mode

## Usage Guidelines

This command can be used to enable PBR globally on all interfaces or on a specific interface.

The **no** form of the command disables PBR.

## Examples

The following example enables PBR globally.

```
device(config)# route-map map1
device(config-routemap map1)# exit
device(config)# ip policy route-map map1
```

The following example enables PBR on a specific interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip policy route-map map1
```

## History

| Release version | Command history |
|---|---|
| 8.0.40a | Support for this command was added for the Ruckus ICX 7250. |

# ip prefix-list

Configures a RIP routing prefix list that can permit or deny specific routes. The prefix list can be applied globally or to individual interfaces, where they may apply to incoming (learned) our outgoing (advertised) routes.

## Syntax

**ip prefix-list** *name* [ **seq** *number* ] { **permit** | **deny** } { *source-ip-address / L* }

**ip prefix-list** *name* **description** *string*

**no ip prefix-list** *name* [ **seq** *number* ] { **permit** | **deny** } { *source-ip-address / L* }

**no ip prefix-list** *name* **description** *string*

## Command Default

By default, routes that do not match a prefix list are learned or advertised. To prevent a route from being learned or advertised, you must configure and apply a prefix list to deny the route.

## Parameters

*name*

Identifies the prefix list.

**description** *string*

Provides information describing the named prefix list in an ASCII string.

**seq** *number*

Specifies an optional sequence number for the named prefix list.

**permit**

Indicates that designated routes will be allowed; that is, either learned or advertised, depending on how the prefix list is applied.

**deny**

Indicates that designated routes will be denied; that is, will not be learned or will not be advertised, depending on how the prefix list is applied.

*source-ip-address / L*

Designates a specific route, based on its IP address prefix and mask length.

[ **ge** *value* ] [ **le** *value* **]**

The keyword **le** indicates the maximum prefix length that can be matched. The keyword **ge** indicates minimum prefix length that can match. Possible values for ge (greater than or equal to) and le (less than or equal to) are 1 through 32. The **ge** and **le** values can be used separately or together.

## Modes

Global configuration mode

# Usage Guidelines

The **no** form of the command disables the prefix list.

A route is defined by the destination's IP address and network mask. Because the default action is permit, all other routes (routes not explicitly permitted or denied by the filters) can be learned or advertised.

Prefix lists can be applied to RIP globally using the separate **prefix-list** command or at the interface level using the separate **ip rip prefix-list** command.

# Examples

The following example creates four prefix lists. Three of the prefix lists permit a route for a different network. The last prefix list denies a route for one network. The routes are defined but not applied in the example.

```
device# configure terminal
device(config)# ip prefix-list list1 permit 10.53.4.1 255.255.255.0
device(config)# ip prefix-list list2 permit 10.53.5.1 255.255.255.0
device(config)# ip prefix-list list3 permit 10.53.6.1 255.255.255.0
device(config)# ip prefix-list list4 deny 10.53.7.1 255.255.255.0
```

# ip preserve-acl-user-input-format

Preserves the user input format for ACL configuration.

## Syntax

**ip preserve-acl-user-input-format**

**no ip preserve-acl-user-input-format**

## Command Default

ACL implementations automatically display the TCP or UDP port name instead of the port number.

## Modes

Global configuration mode

## Usage Guidelines

When the option to preserve user input is enabled, the system displays either the port name or the number as used during configuration.

The **no** form of the command removes the user input perseverance configuration.

# Examples

The following example shows the behavior when the option to preserve user input is enabled. In this example, the TCP port is configured by number (80) when configuring ACL group 140. However, **show ip access-lists 140** reverts to the port name for the TCP port (HTTP in this example). When the **ip preserve-acl-user-input-format** command is configured, the **show ip access-lists** command displays either the TCP port number or name, depending on how it was configured by the user.

```
device(config)# access-list 140 permit tcp any any eq 80
device(config)# access-list 140 permit tcp any any eq ftp
device(config)# exit

device# show ip access-lists 140
Extended IP access list 140
permit tcp any any eq http
permit tcp any any eq ftp

device(config)# access-list 140 permit tcp any any eq 80
device(config)# access-list 140 permit tcp any any eq ftp

device# show ip access-lists 140
Extended IP access list 140
permit tcp any any eq http
permit tcp any any eq ftp

device(config)# ip preserve-acl-user-input-format
device(config)# exit

device# show ip access-lists 140
Extended IP access list 140
permit tcp any any eq 80
permit tcp any any eq ftp
```

# ip-proto

Configures an IP protocol-based VLAN.

## Syntax

**ip-proto** [ **name** *string* ]

**no ip-proto** [ **name** *string* ]

## Command Default

An IP protocol-based VLAN is not configured.

## Parameters

**name** *string*

Specifies the name of the IP protocol VLAN. The maximum length of the string is 32 characters.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the IP protocol-based VLAN.

## Examples

The following example configures the IP protocol-based VLAN.

```
device (config)# vlan 10
device(config-vlan-10)# ip-proto name IP_Prot_VLAN
```

# ip proxy-arp

Enables IP proxy ARP globally.

## Syntax

**ip proxy-arp**

**no ip proxy-arp**

## Command Default

Proxy ARP is disabled by default on Layer 3 switches.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

Proxy ARP allows a Layer 3 switch to answer ARP requests from devices on one network on behalf of devices in another network. Because ARP requests are MAC-layer broadcasts, they reach only the devices that are directly connected to the sender of the ARP request. Thus, ARP requests do not cross routers.

An ARP request from one subnet can reach another subnet when both subnets are on the same physical segment (Ethernet cable), because MAC-layer broadcasts reach all the devices on the segment.

This feature is not supported on Layer 2 switches.

The **no** form of the command disables IP proxy ARP.

## Examples

The following example enables IP proxy ARP globally.

```
device(config)# ip proxy-arp
```

The following example enables proxy ARP on port 1/2/1.

```
device# configure terminal
device(config)# interface ethernet 1/2/1
device(config-if-e1000-1/2/1)# ip proxy-arp
```

# ip proxy-arp (interface)

Enables IP proxy ARP on an interface.

## Syntax

**ip proxy-arp** { **enable** | **disable** }

**no ip proxy-arp** { **enable** | **disable** }

## Command Default

IP proxy ARP is disabled.

## Parameters

**enable**

Enables IP proxy ARP on an interface.

**disable**

Disables IP proxy ARP on an interface.

## Modes

Interface configuration mode

## Usage Guidelines

Configuring proxy ARP at the Interface level overrides the global configuration.

Proxy ARP allows a Layer 3 switch to answer ARP requests from devices on one network on behalf of devices in another network. Because ARP requests are MAC-layer broadcasts, they reach only the devices that are directly connected to the sender of the ARP request. Thus, ARP requests do not cross routers.

An ARP request from one subnet can reach another subnet when both subnets are on the same physical segment (Ethernet cable), because MAC-layer broadcasts reach all the devices on the segment.

This feature is not supported on Ruckus Layer 2 switches.

The **no** form of the command enables or disables IP proxy ARP.

## Examples

The following example enables IP proxy ARP on an interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip proxy-arp enable
```

The following example disables IP proxy ARP on an interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip proxy-arp disable
```

# ip radius source-interface

Configures an interface as the source IP address from which the RADIUS client sends RADIUS requests or receives responses.

## Syntax

**ip radius source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

**no ip radius source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

## Command Default

When a management VRF is configured, the RADIUS client sends RADIUS requests and receives responses only through ports belonging to the management VRF and through the out-of-band management port.

## Parameters

**ethernet** *stack-id/slot/port*
Specifies the Ethernet interface address used for setting the source IP address.

**loopback** *number*
Specifies the loopback interface address used for setting the source IP address.

**management** *number*
Specifies the management interface address used for setting the source IP address.

**ve** *number*
Specifies the Virtual Ethernet interface address used for setting the source IP address.

## Modes

Global configuration mode

## Usage Guidelines

When a source interface is configured, management applications use the lowest configured IP address of the specified interface as the source IP address in all the outgoing packets. If the configured interface is not part of the management VRF, the response packet does not reach the destination.

The RADIUS source interface configuration command **ip radius source-interface** should be compatible with the management VRF configuration.

**NOTE**
Any change in the management VRF configuration takes effect immediately for the RADIUS client.

The **no** form of the command removes the configured interface as the source IP address for the RADIUS client.

# Examples

The following example configures an Ethernet interface as the source IP address for the RADIUS client.

```
device(config)# ip radius source-interface ethernet 1/1/1
```

The following example configures a loopback interface as the source IP address for the RADIUS client.

```
device(config)# ip radius source-interface loopback 1
```

# ip rarp

Enables IP Reverse Address Resolution Protocol (RARP).

## Syntax

**ip rarp**

**no ip rarp**

## Command Default

RARP is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

RARP allows an IP host that does not have a means of storing its IP address across power cycles or software reloads to query a directly-attached router for an IP address. RARP is enabled by default. However, you must create a RARP entry for each host that will use the Layer 3 switch for booting.

The **no** form of the command disables IP RARP.

## Examples

The following example disables IP RARP.

```
device(config)# no ip rarp
```

# ip redirect

Enables IPv4 redirect messages on individual Virtual Ethernet (VE) interface.

## Syntax

**ip redirect**

**no ip redirect**

## Command Default

Redirect is not enabled.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables IPv4 redirect messages.

**NOTE**
The command is supported only on VE interface configuration mode.

## Examples

The following example enables ICMP redirect on a VE interface.

```
device(config-vlan-10)# interface ve 10
device(config-vif-10)# ip redirect
```

# ip rip

Configures Routing Information Protocol at the interface level. RIP must first be enabled globally on the device.

## Syntax

**ip rip** { **v1-only** | **v1-compatible-v2** | **v2-only** }

**no ip rip** { **v1-only** | **v1-compatible-v2** | **v2-only** }

## Command Default

By default, RIP is not configured on any interface.

## Parameters

**v1-only**
Configures the interface for RIP Version.

**v1-compatible-v2**
Configures the interface for RIP Version 1 with RIP Version 2 compatibility.

**v2-only**
Configures the interface for RIP Version 2.

## Modes

Interface configuration mode.

## Usage Guidelines

The **no** form of the command disables RIP on the interface.

RIP must first be configured globally. Refer to the **router rip** command. Then you must configure individual interfaces, including physical interfaces as well as virtual routing interfaces, with the **ip rip** command.

## Examples

The following example configures RIP Version 1 on Ethernet interface 1/2/3 (device 1/slot 2/interface 3).

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e01000-1/2/3)# ip rip v1-only
```

The following examples removes RIP configuration from the same interface.

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e01000-1/2/3)# no ip rip v1-only
```

# ip rip metric-offset

Increases the cost metric an interface applies to learned or advertised RIP routes.

## Syntax

**ip rip metric-offset** *num* { **in** | **out** }

**no ip rip metric-offset** *num* { **in** | **out** }

## Command Default

By default, the interface adds one to the route metric before storing the route.

## Parameters

*num*

A decimal number from 1 through 16 that the interface adds to the cost metric for learned or advertised RIP routes.

**in**

Applies cost to routes the interface learns from RIP neighbors.

**out**

Applies cost to routes the interface advertises to RIP neighbors.

## Modes

Interface configuration mode.

## Usage Guidelines

The **no** form of the command removes the added cost from RIP routes learned or advertised on the interface.

Routes with a higher cost are less likely to be used. You can prevent the RIP router from using a route learned on a particular interface by adding a cost metric of 16 on the interface.

## Examples

The following example adds 5 to the cost metric for routes advertised on Ethernet interface 1/2/3 (Device 1/Slot 2/ Interface 3).

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e1000-1/2/3)# ip rip metric-offset 5 out
```

The following example returns the advertised route metric to default (1) for the interface in the previous example.

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e1000-1/2/3)# no ip rip metric-offset 5 out
```

The following example prevents the RIP router from using RIP routes learned on Ethernet interface 1/2/3.

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e1000-1/2/3)# ip rip metric-offset 16 in
```

# ip rip prefix-list

Applies a pre-configured prefix-list to a RIP interface.

## Syntax

**ip rip prefix-list** *name* { **in** | **out** }

**no ip rip prefix-list** *name* { **in** | **out** }

## Command Default

By default, all routes are learned from and advertised to RIP neighbors.

## Parameters

*name*

Designates the RIP prefix list to be applied.

**in**

Applies the designated prefix list as a filter to incoming routes; that, is to routes learned from RIP neighbors.

**out**

Applies the designated prefix list as a filter to outgoing routes; that, is to routes advertised to RIP neighbors.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command removes the prefix list from the interface.

Prefix lists must be configured with the **ip prefix-list** command before they are applied.

Prefix lists can be applied globally with the **prefix-list** command.

## Examples

The following example applies the prefix list named list2 to RIP routes learned on Ethernet interface 1/1/2 and a different prefix list, list 3, to RIP routes advertised on the same interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/2
device(config-if-e1000-1/1/2)# ip rip prefix-list list2 in
device(config-if-e1000-1/1/2)# ip rip prefix-list list3 out
```

# ip rip route-map

Applies a pre-configured route map to a RIP interface.

## Syntax

**ip rip route-map** *name* { **in** | **out** }

**no ip rip route-map**

## Parameters

*name*

Specifies the route-map to be applied.

**in**

Applies the route-map as an inbound filter; that is, it applies to routes learned from RIP neighbors.

**out**

Applies the route-map as an outbound filter; that is, it applies to routes advertised to RIP neighbors.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command removes the route-map from the interface.

An access control list (ACL) or a prefix list can be applied as a route-map using this command.

## Examples

The following command applies the route-map named map1 to filter RIP routes learned on Ethernet interface 1/1/2.

```
device# configure terminal
device(config)# interface ethernet 1/1/2
device(config-if-e1000-1/1/2)# ip rip route-map map1 in
```

# ip route

Adds a static route to the IP routing tables.

## Syntax

**ip route** [ **vrf** *vrf-name* ] *dest-ip-addr next-hop-addr* [ *metric* ] [ **distance** *distance* ] [ **tag** *tag* ] [ **name** *string* ]

**ip route** [ **vrf** *vrf-name* ] *dest-ip-addr* { **ethernet** *unit***/***slot***/***port* | **ve** *number* | **tunnel** *tunnel-id* } [ *metric* ] [ **name** *string* ]

**ip route** [ **vrf** *vrf-name* ] *dest-ip-addr* **null0** [ **name** *string* ]

**no ip route** [ **vrf** *vrf-name* ] *dest-ip-addr next-hop-addr* [ *metric* ] [ **distance** *distance* ] [ **tag** *tag* ] [ **name** *string* ]

**no ip route** [ **vrf** *vrf-name* ] *dest-ip-addr* { **ethernet** *unit***/***slot***/***port* | **ve** *number* | **tunnel** *tunnel-id* } [ *metric* ] [ **name** *string* ]

**no ip route** [ **vrf** *vrf-name* ] *dest-ip-addr* **null0** [ **name** *string* ]

## Parameters

**vrf** *vrf-name*
Specifies the VRF associated with the destination IPv4 address.

*dest-ip-addr*
Specifies the destination IPv4 address and mask in the format A.B.C.D/L (where "L" is the prefix-length of the mask) or A.B.C.D P.Q.R.S (where "PQ.R.S" is the mask value).

*next-hop-addr*
Specifies the IPv4 address of the next hop.

**ethernet** *unit***/***slot***/***port*
Specifies the destination Ethernet port.

**ve** *vlan-id*
Specifies the outgoing interface type as VE.

**tunnel** *tunnel-id*
Specifies the outgoing interface type as tunnel.

**null0**
Configures the Layer 3 switch to drop IP packets to a specific network or host address by configuring a "null" (sometimes called "null0") static route for the address.

*metric*
Specifies the cost metric of the route. Valid values range from 1 through 16. The default is 1.

**distance** *distance*
Specifies the administrative distance of the route. When comparing otherwise equal routes to a destination, a device prefers lower administrative distances over higher ones. Valid values range from 1 through 255. The default is 1. The value 255 makes the route unusable.

**tag** *tag*
Specifies the tag value of the route to use for route filtering with a route map. Valid values range from 0 through 4294967295. The default is 0.

**name** *string*
Specifies the static route name.The maximum length of the name is 128 bytes.

# Modes

Global configuration mode

# Usage Guidelines

The **no** form of the command followed by the route identifier removes a static route. If the static route includes a name, you must enter the **no** form of the command twice (once to remove the name and the second time to remove the route from the routing table).

For a default route, enter 0.0.0.0/0 as the destination IP address followed by the next-hop IP address. Physical or virtual interfaces cannot be used as next-hops for a default route.

If you do not want to specify a next-hop IP address, you can instead specify a port or interface number on the device. If you specify an Ethernet port, the device forwards packets destined for the static route's destination network to the specified interface. Conceptually, this feature makes the destination network like a directly connected network, associated with a device interface.

The port or virtual interface you use for the static route's next hop must have at least one IP address configured on it. The address does not need to be in the same subnet as the destination network.

ICX 7150 devices do not support tunnels or VRFs.

When a tunnel is configured as the next hop for a static route, the tunnel must already be configured if the destination is a non-default VRF. In contrast, a tunnel can be designated as the next hop in the default VRF before it is configured. The default VRF is used when no VRF is specified in the command.

# Examples

The following example configures a static route to 10.95.7.0, using 10.95.6.157 as the next-hop gateway.

```
device(config)# ip route 10.95.7.0/24 10.95.6.157
```

The following example configures a default route through next-hop IP address 10.2.12.1.

```
device(config)# ip route 0.0.0.0/0 10.2.12.1
```

The following example configures a static route with an Ethernet interface as the destination.

```
device(config)# ip route 192.128.2.69 255.255.255.0 ethernet 1/4/1
```

The following example configures a null static route to drop packets destined for network 10.157.22.x.

```
device(config)# ip route 10.157.22.0 255.255.255.0 null0
```

The following example configures tunnel 1 as the next hop gateway. The tunnel is configured in the default VRF.

```
device(config)# ip route 56.1.1.0/24 tunnel 1
```

The following example configures tunnel 5 as the next hop gateway in a non-default VRF, as long as the tunnel already exists.

```
device(config)# ip route vrf 1 56.1.5.0/24 tunnel 5
```

# ip route next-hop

Enables a device to use routes from a specified protocol to resolve a configured static route.

## Syntax

**ip route next-hop** { **bgp** | **ospf** | **rip** }

**no ip next-hop** { **bgp** | **ospf** | **rip** }

## Command Default

The device is not enabled to use routes from a specified protocol to resolve a configured static route.

## Parameters

**bgp**

Configures the device to use iBGP and eBGP routes to resolve static routes.

**ospf**

Configures the device to use OSPF routes to resolve static routes.

**rip**

Configures the device to use RIP routes to resolve static routes.

## Modes

Global configuration mode

## Usage Guidelines

This command can be independently applied on a per-VRF basis. Connected routes are always used to resolve static routes.

The **no** form of the command disables the device to use routes from a specified protocol to resolve static routes.

## Examples

The following example configures the device to use OSPF protocol to resolve static routes.

```
device(config)# ip route next-hop ospf
```

# ip route next-hop-enable-default

Enables a device to use the default route (0.0.0.0/0) to resolve a static route.

## Syntax

**ip route next-hop-enable-default**

**no ip route next-hop-enable-default**

## Command Default

Device does not use the default route to resolve static route.

## Modes

Global configuration mode

## Usage Guidelines

This command can be independently applied on a per-VRF basis.

This command works independently with the ip route next-hop-recursion and ip route next-hop commands. If the default route is a protocol route, that protocol needs to be enabled to resolve static routes using the **ip route next-hop** command in order for static routes to resolve by this default route. If the default route itself is a static route, you must configure the **ip route next-hop-recursion** command to resolve other static routes by this default route.

The **no** form of the command disables the device to use the default route to resolve a static route.

## Examples

The following example enables static route resolve by default route

```
device(config)# ip route next-hop-enable-default
```

# ip route next-hop-recursion

Enables a device to use static routes to resolve another static route.

## Syntax

**ip route next-hop-recursion** [ *level* ]

**no ip route next-hop-recursion** *level*

## Command Default

The recursive static route next hop lookup is disabled.

## Parameters

*level*

Specifies the number of levels of recursion allowed. Valid values are 1 to 10. The default value is 3.

## Modes

Global configuration mode

## Usage Guidelines

This command can be independently applied on a per-VRF basis..

The **no** form of the command disables the recursive static route next hop lookup.

## Examples

The following example enables the device to use static routes to resolve another static route.

```
device(config)# ip route next-hop-recursion 5
```

# ip router-id

Configures an IPv4 router ID.

## Syntax

**ip router-id** *ipv4-address*

**no ip router-id** *ipv4-address*

## Command Default

A router ID is not configured.

## Parameters

*ipv4-address*

Specifies the IPv4 address. The default is the lowest IP address in use.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the configured IPv4 router ID.

## Examples

The following example configures the IPv4 router ID.

```
device(config)# ip router-id 10.14.52.11
```

# ip show-portname

Displays interface names in syslog messages.

## Syntax

**ip show-portname**

**no ip show-portname**

## Command Default

An interface slot number (if applicable), port number, and interface type are displayed when you display syslog messages.

## Modes

Global configuration mode

## Usage Guidelines

Syslog messages show the interface type, such as "ethernet", and so on. However, if the **ip show-portname** command is configured and a name has been assigned to the port, the port name replaces the interface type.

The **no** form of the command displays the interface slot number and port number in syslog messages.

## Examples

The following example configures the display of the interface name in syslog messages.

```
device(config)# ip show-portname
```

# ip show-service-number-in-log

Displays TCP or UDP port numbers instead of the port names.

## Syntax

**ip show-service-number-in-log**

**no ip show-service-number-in-log**

## Command Default

By default, the device displays TCP or UDP application information in named notation.

## Modes

Global configuration mode

## Usage Guidelines

When this command is enabled, the device displays 80 (the port number) instead of http (the well-known port name) in the output of show commands and other commands that contain application port information.

The **no** form of the command displays the TCP or UDP port name.

## Examples

The following example sets the display of TCP or UDP port numbers instead of their names.

```
device(config)# ip show-service-number-in-log
```

# ip show-subnet-length

Enables CIDR format for displaying network masks.

## Syntax

**ip show-subnet-length**

**no ip show-subnet-length**

## Command Default

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0).

## Modes

Global configuration mode

## Usage Guidelines

This command changes the displays to prefix format (CIDR format) (example: /18) on a Layer 3 switch or Layer 2 switch.

The **no** form of the command enables the display of network masks in classical IP address format.

## Examples

The following example enables CIDR format for displaying network masks.

```
device(config)# ip show-subnet-length
```

# ip source-route

Enables forwarding of IP source-routed packets.

## Syntax

**ip source-route**

**no ip source-route**

## Command Default

The Layer 3 switch forwards both types of source-routed (strict and loose) packets by default.

## Modes

Global configuration mode

## Usage Guidelines

A source-routed packet specifies the exact router path for the packet. The packet specifies the path by listing the IP addresses of the router interfaces through which the packet must pass on its way to the destination. The Layer 3 switch supports strict and loose types of IP source routing.

Strict source routing requires the packet to pass through only the listed routers. If the Layer 3 switch receives a strict source-routed packet but cannot reach the next hop interface specified by the packet, the Layer 3 switch discards the packet and sends an ICMP Source-Route-Failure message to the sender.

**NOTE**
The Layer 3 switch allows you to disable sending of the Source-Route-Failure messages.

Loose source routing requires that the packet pass through all of the listed routers but also allows the packet to travel through other routers, which are not listed in the packet.

The **no** form of the command disables forwarding of IP source-routed packets.

## Examples

The following example disables forwarding of IP source-routed packets.

```
device# configure terminal
device(config)# no ip source-route
```

The following example reenables forwarding of IP source-routed packets.

```
device(config)# ip source-route
```

# ip ssh authentication-retries

Configures the number of SSH authentication retries.

## Syntax

**ip ssh authentication-retries** *number-retries*

**no ip ssh authentication-retries** *number-retries*

## Command Default

By default, the device attempts to negotiate a connection with the connecting host three times.

## Parameters

*number-retries*

The number of SSH authentication retries. Valid values are from 1 through 5.

## Modes

Global configuration mode

## Usage Guidelines

The **ip ssh authentication-retries** command is not applicable on devices that act as an SSH client. On such devices, when you try to establish an SSH connection with the wrong credentials, the session is not established and the connection is terminated. The device does not check the SSH authentication retry configuration set using the **ip ssh authentication-retries** command. The command is applicable only to SSH clients such as PuTTY, SecureCRT, and so on.

The **no** form of the command sets the number of retries to the default value of three.

## Examples

The following example shows how to set the authentication retries to 5.

```
device(config)# ip ssh authentication-retries 5
```

# ip ssh client

Restricts Secure Shell (SSH) access to a device based on the client IP address and MAC address.

## Syntax

**ip ssh client** { *ipv4-address* [ *mac-address* ] | **any** *mac-address* | **ipv6** *ipv6-address* }

**no ip ssh client** { *ipv4-address* [ *mac-address* ] | **any** *mac-address* | **ipv6** *ipv6-address* }

## Command Default

SSH access is not enabled.

## Parameters

*ipv4-address*
: Allows SSH access from the host with the specified IP address.

*mac-address*
: Allows SSH access from the host with the specified IP address and MAC address.

**any** *mac-address*
: Allows SSH access from any host with any IP address and specified MAC address.

**ipv6** *ipv6-address*
: Allows SSH access from any host with the specified IPv6 address.

## Modes

Global configuration mode

## Usage Guidelines

You can specify only one IP address with one command. However, you can enter the command ten times to specify up to ten IP addresses.

The **no** form of the command removes the SSH access restrictions.

## Examples

The following example shows how to allow SSH access to a device based on the host with IP address 10.157.22.39.

```
device(config)# ip ssh client 10.157.22.39
```

The following example shows how to allow SSH access to the device based on the host with IP address 10.157.22.39 and MAC address 0000.000f.e9a0.

```
device(config)# ip ssh client 10.157.22.39 0000.000f.e9a0
```

The following example shows how to allow SSH access to the device based on the host with IPv6 address 2001::1 and MAC address 0000.000f.e9a0.

```
device(config)# ip ssh client ipv6 2001::1
```

# ip ssh encryption aes-only

Enables SSH AES encryption and disables support for 3des-cbc.

## Syntax

**ip ssh encryption aes-only**

**no ip ssh encryption aes-only**

## Command Default

The 3des-cbc encryption is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables the AES encryption support.

## Examples

The following example shows how to enable AES encyrption.

```
device(config)# ip ssh encryption aes-only.
```

# ip ssh encryption disable-aes-cbc

Disables the Advanced Encryption Standard - Cipher-Block Chaining (AES-CBC) encryption mode for the Secure Shell (SSH) protocol.

## Syntax

**ip ssh encryption disable-aes-cbc**

**no ip ssh encryption disable-aes-cbc**

## Command Default

If JITC is enabled, only AES-CTR encryption mode is supported and AES-CBC mode is disabled by default. In the standard mode, the AES-CBC encryption mode is enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command enables the AES-CBC encryption mode.

## Examples

The following example disables the AES-CBC encryption mode.

```
device(config)#  ip ssh encryption disable-aes-cbc
```

## History

| Release version | Command history |
|---|---|
| 08.0.20a | This command was introduced. |

# ip ssh idle-time

Configures the amount of time an SSH session can be inactive before the device closes it.

## Syntax

**ip ssh idle-time** *time*

**no ip ssh idle-time** *time*

## Command Default

By default, SSH sessions do not time out.

## Parameters

*time*

Time in minutes. Valid values are from 0 through 240. The default is 0 (never time out).

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the timeout value to the default.

If an established SSH session has no activity for the specified number of minutes, the device closes it. An idle time of 0 minutes (the default value) means that SSH sessions never time out.

## Examples

The following example configures the SSH idle time to 50 minutes.

```
device(config)# ip ssh idle-time 50
```

# ip ssh interactive-authentication

Configures the keyboard-interactive authentication.

## Syntax

**ip ssh interactive-authentication** { **yes** | **no** }

**no ip ssh interactive-authentication** { **yes** | **no** }

## Command Default

Keyboard-interactive authentication is not enabled.

## Parameters

**yes**

Enables keyboard-interactive authentication.

**no**

Disables keyboard-interactive authentication.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables keyboard-interactive authentication.

## Examples

The following example enables keyboard-interactive authentication.

```
device(config)# ip ssh interactive-authentication yes
```

# ip ssh key-authentication

Configures DSA or RSA challenge-response authentication.

## Syntax

**ip ssh key-authentication** { **yes** | **no** }

**no ip ssh key-authentication** { **yes** | **no** }

## Command Default

DSA or RSA challenge-response authentication is enabled by default.

## Parameters

**yes**

Enables DSA or RSA challenge-response authentication. The default is **yes**.

**no**

Disables DSA or RSA challenge-response authentication.

## Modes

Global configuration mode

## Usage Guidelines

After the SSH server on the device negotiates a session key and encryption method with the connecting client, user authentication takes place. The implementation of SSH supports DSA or RSA challenge-response authentication and password authentication. You can deactivate one or both user authentication methods for SSH. Note that deactivating both authentication methods disables the SSH server entirely.

With DSA or RSA challenge-response authentication, a collection of clients' public keys are stored on the device. Clients are authenticated using these stored public keys. Only clients that have a private key that corresponds to one of the stored public keys can gain access to the device using SSH.

The **no** form of the command disables DSA or RSA challenge-response authentication.

## Examples

The following example enables DSA or RSA challenge-response authentication.

```
device(config)# ip ssh key-authentication
```

# ip ssh key-exchange-method dh-group1-sha1

Disables or re-enables diffie-hellman-group1-sha1 as the key-exchange method to establish an SSH connection.

## Syntax

**ip ssh key-exchange-method dh-group1-sha1**

**no ip ssh key-exchange-method dh-group1-sha1**

## Command Default

The diffie-hellman-group14-sha1 is used as the default key-exchange method. When the user disables diffie-hellman-group1-sha1 as the key-exchange method, the diffie-hellman-group14-sha1 takes over as the key-exchange method.

## Modes

Global configuration mode

## Usage Guidelines

The **ip ssh key-exchange-method dh-group1-sha1** command is not supported in FIPS or CC mode.

The **no** form of the command disables diffie-hellman-group1-sha1 as the key-exchange method.

In FIPS mode, only diffie-helman-group-exchange-sha256 is supported and in common criteria(CC) mode, only diffie-hellman-group14-sha1 is supported.

## Examples

The following example disables diffie-hellman-group1-sha1 as the key-exchange method.

```
device(config)# no ip ssh key-exchange-method dh-group1-sha1
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# ip ssh password-authentication

Configures password authentication.

## Syntax

**ip ssh password-authentication** { **yes** | **no** }

**no ip ssh password-authentication** { **yes** | **no** }

## Command Default

Password authentication is enabled.

## Parameters

**yes**

Enables the password authentication. The default is **yes**.

**no**

Disables the password authentication.

## Modes

Global configuration mode

## Usage Guidelines

After the SSH server on the device negotiates a session key and encryption method with the connecting client, user authentication takes place. The implementation of SSH supports DSA or RSA challenge-response authentication and password authentication. You can deactivate one or both user authentication methods for SSH. Note that deactivating both authentication methods disables the SSH server entirely.

With password authentication, users are prompted for a password when they attempt to log in to the device (provided empty password logins are not allowed). If there is no user account that matches the username and password supplied by the user, the user is not granted access.

The **no** form of the command enables password authentication.

## Examples

The following example disables the password authentication.

```
device(config)# ip ssh password-authentication yes
```

# ip ssh permit-empty-password

Allows a user with an SSH client to log in without being prompted for a password.

## Syntax

**ip ssh permit-empty-password** { **yes** | **no** }

**no ip ssh permit-empty-password** { **yes** | **no** }

## Command Default

By default, empty password logins are not allowed.

## Parameters

**yes**

Allows a user to log in to an SSH client without being prompted for a password.

**no**

Disallows a user to log in to an SSH client without being prompted for a password.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disallows user to log in without being prompted for a password.

By default, empty password logins are not allowed; users with an SSH client are always prompted for a password when they log in to the device. To gain access to the device, each user must have a username and password. Without a username and password, a user is not granted access.

If you enable empty password logins, users are not prompted for a password when they log in. Any user with an SSH client can log in without being prompted for a password.

## Examples

The following example enables the user to log in to an SSH client without being prompted for a password.

```
device(config)# ip ssh permit-empty-password yes
```

# ip ssh port

Configures the port for SSH traffic.

## Syntax

**ip ssh port** *port-num*

**no ip ssh port** *port-num*

## Command Default

By default, SSH traffic occurs on TCP port 22.

## Parameters

*port-num*
Specifies the port number.

## Modes

Global configuration mode

## Usage Guidelines

If you change the default SSH port number, you must configure SSH clients to connect to the new port. Also, you should be careful not to assign SSH to a port that is used by another service. If you change the SSH port number, Ruckus recommends that you change it to a port number greater than 1024.

The **no** form of the command changes the port to the default.

## Examples

The following example configures the SSH port as 2200.

```
device(config)# ip ssh port 2200
```

# ip ssh pub-key-file

Imports the authorized public keys into the active configuration of the device by loading the public key file from a TFTP server.

## Syntax

**ip ssh pub-key-file** { **remove** | **tftp** { *ipv4-address* | **ipv6** *ipv6-address*} *file-name* }

**no ip ssh pub-key-file** { **remove** | **tftp** { *ipv4-address* | **ipv6** *ipv6-address*} *file-name* }

## Command Default

The private key is normally stored in a password-protected file on the local host; the public key is stored in another file and is not protected.

## Parameters

**remove**
Removes the SSH client public key file from the device.

**tftp**
Imports DSS public key from the TFTP server.

*ipv4-address*
Specifies the IPv4 address of the TFTP server.

**ipv6** *ipv6-address*
Specifies the IPv6 address of the TFTP server.

*file-name*
Specifies the public key file name.

## Modes

Global configuration mode

## Usage Guidelines

You can use the **show ip client-pub-key** command to display the currently loaded public keys.

SSH clients that support DSA or RSA authentication normally provide a utility to generate a DSA or RSA key pair. The private key is normally stored in a password-protected file on the local host; the public key is stored in another file and is not protected. You must import the client public key for each client into the Ruckus device.

The **no** form of the command removes the imported public keys.

# Examples

The following example imports a public key file from the TFTP server 192.168.10.1.

```
device(conig)# ip ssh pub-key-file tftp 192.168.10.1 pkeys.txt
```

The following example removes a public key file from the device.

```
device(confg)# ip ssh pub-key-file remove
```

# ip ssh rekey

Configures the Secure Shell (SSH) rekey interval, either in terms of the maximum number of minutes or the maximum amount of data.

## Syntax

**ip ssh rekey** { **client** | **server** } { **data** *Kbytes* | **time** *minutes* }

**no ip ssh rekey** { **client** | **server** } { **data** *Kbytes* | **time** *minutes* }

## Command Default

SSH rekey is disabled by default in non-FIPS mode. In FIPS/CC mode the default value for time and data is 30 minutes and 500000 KB respectively. SSH rekey feature cannot be disabled in FIPS/CC mode.

## Parameters

**client**

Specifies the rekey interval for the SSH client sessions.

**server**

Specifies the rekey interval for the SSH server sessions.

**data** *KBytes*

Configures the maximum amount of data (in kilobytes) that can be transmitted before SSH rekey is initiated. The valid range is from 100 through 2000000 kilobytes.

**time** *minutes*

Configures the maximum time in minutes before SSH rekey is initiated. The valid range is from 0 through 120 minutes.

## Modes

Global configuration mode

## Usage Guidelines

When the value for *minutes* is set to 0, SSH rekey does not take place.

It is recommended that the rekey data value not be configured higher than one Gigabyte.

In FIPS or CC mode, SSH rekey is enabled by default and cannot be disabled. The default value for time is 30 minutes and the default value for data is 500 MB in both FIPS and CC mode. If the rekey configuration is removed, the default values are applied. The default values are not displayed in the configuration.

Non-FIPS mode to FIPS or CC mode: If the rekey configuration is configured in non-FIPS mode, the same values are applied while moving to FIPS mode. If the rekey configuration is not configured in non-FIPS mode, the default values in FIPS or CC mode will be applied.

FIPS or CC mode to non-FIPS mode: The configuration in FIPS mode or CC mode is removed, and SSH rekey is disabled while moving to non-FIPS mode.

The **no** form of the command disables SSH rekey in normal operating mode.

# Examples

The following example configures SSH rekey on the outbound SSH session every hour.

```
device# configure terminal
device(config)# ip ssh rekey client time 60
```

The following example configures SSH rekey on the inbound SSH session whenever 10000 kilobytes of data has been transmitted.

```
device# configure terminal
device(config)# ip ssh rekey server data 10000
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# ip ssh scp

Enables Secure Copy (SCP).

## Syntax

**ip ssh scp** { **enable** | **disable** }

**no ip ssh scp** { **enable** | **disable** }

## Command Default

SCP is enabled.

## Parameters

**enable**

Enables SCP.

**disable**

Disables SCP.

## Modes

Global configuration mode

## Usage Guidelines

SCP uses security built into SSH to transfer image and configuration files to and from the device. SCP automatically uses the authentication methods, encryption algorithm, and data compression level configured for SSH.

If you disable SSH, SCP is also disabled.

The **no** form of the command disables SCP.

## Examples

The following example disables SCP.

```
device(config)# ip ssh scp disable
```

The following example enables SCP.

```
device(config)# ip ssh scp enable
```

# ip ssh strict-management-vrf

Allows incoming SSH connection requests only from the management VRF and not from the out-of-band (OOB) management port.

## Syntax

**ip ssh strict-management-vrf**

**no ip ssh strict-management-vrf**

## Command Default

When the management VRF is configured, incoming SSH connection requests are allowed from the ports that belong to the management VRF and from the OOB management port.

## Modes

Global configuration mode

## Usage Guidelines

The **ip ssh strict-management-vrf** command is applicable only when the management VRF is configured. If a management VRF is not configured, configuring the **ip ssh strict-management-vrf** command displays an error message.

The **ip ssh strict-management-vrf** command does not prevent a connection initiated from the OOB management interface if the management interface VRF and the management VRF are the same. The user must configure either the **management exclude all oob** command or the **management exclude ssh oob** command.

For the SSH server, changing the management VRF configuration or configuring the **ip ssh strict-management-vrf** command does not affect the existing SSH connections. The changes are applied only to new incoming connection requests.

The **ip ssh strict-management-vrf** command and the **management exclude** commands are mutually exclusive. If the latter command is configured, outbound SSH connections are not blocked.

The **no** form of the command enables the incoming SSH connection requests from ports that belong to the management VRF and from the OOB management port.

## Examples

The following example allows incoming SSH connection requests from the management VRF only.

```
device(config)# ip ssh strict-management-vrf
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | The Usage Guidelines were modified. |

# Related Commands

management exclude

# ip ssh timeout

Configures the wait time for a response from the client when the SSH server attempts to negotiate a session key and encryption method with a connecting client.

## Syntax

**ip ssh timeout** *time*

**no ip ssh timeout** *time*

## Command Default

The default timeout value is 120 seconds.

## Parameters

*time*

Timeout value in seconds. The valid range is from 1 through 120 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the timeout value to the default.

## Examples

The following example configures the SSH timeout value to 60 seconds.

```
device(config)# ip ssh timeout 60
```

# ip ssl

Configures Secure Socket Layer (SSL) settings.

## Syntax

**ip ssl cert-key-size** *size*

**no ip ssl cert-key-size** *size*

**ip ssl** { **certificate-data-file** | **client-certificate** | **client-private-key** | **private-key-file** } **tftp** { *ipv4-address* | **ipv6** *ipv6-address*} *file-name*

**no ip ssl** { **certificate-data-file** | **client-certificate** | **client-private-key** | **private-key-file** } **tftp** { *ipv4-address* | **ipv6** *ipv6-address*} *file-name*

**ip ssl port** *port-num*

**no ip ssl port** *port-num*

**ip ssl certificate** { **common-name** | **country** | **locality** | **org** | **org-unit** | **state** } *name*

**no ip ssl certificate** { **common-name** | **country** | **locality** | **org** | **org-unit** | **state** } *name*

## Command Default

The default key size for Ruckus-issued and imported digital certificates is 2048 bits.

By default, SSL protocol exchanges occur on TCP port 443.

The default TFTP server is not configured.

## Parameters

**cert-key-size** *size*
Configures SSL server certificate key size. Valid values are 2048 and 4096.

**certificate-data-file**
Imports the server RSA certificate.

**client-certificate**
Imports the client RSA certificate.

**client-private-key**
Imports the client RSA private key.

**private-key-file**
Imports the server RSA private key.

**tftp**
Specifies that TFTP is used to import the certificates.

*ipv4-address*
Configures the IPv4 address of the TFTP server from which the certificates are imported.

**ipv6** *ipv6-address*
Configures the IPv6 address of the TFTP server from which the certificates are imported.

*file-name*
The certificate data file name.

**port** *port-num*
Specifies the HTTPS/SSL port. The default port is 443.

**certificate**
Configures the SSL certificate generation signing request.

**common-name**
Specifies the common name, fully qualified domain name, or web address for which you plan to use your certificate.

**country**
Specifies the country name.

**locality**
Specifies the locality name.

**org**
Specifies the organization name.

**org-unit**
Specifies the organization unit name.

**state**
Specifies the state or province name.

*name*
Fully qualified domain name or web address for which you plan to use your certificate (for example, www.server.com) when used with **common-name**, two letter code country name (for example, US) when used with **country**, locality name (for example, city) when used with **locality**, organization name (for example, company) when used with **org**, organization unit name (for example, section) when used with **org-unit**, or province name (for example, California) when used with **state**.

# Modes

Global configuration mode

# Usage Guidelines

The SSL server certificate key size applies only to digital certificates issued by Ruckus and does not apply to imported certificates.

To allow a client to communicate with another Ruckus device using an SSL connection, you configure a set of digital certificates and RSA public-private key pairs on the device. A digital certificate is used for identifying the connecting client to the server. It contains information about the issuing Certificate Authority CA), as well as a public key. You can either import digital certificates and private keys from a server, or you can allow the device to create them. The RSA private key can be up to 4096 bits.

The **no** form of the command removes the configurations.

# Examples

The following example shows how to import a digital certificate issued by a third-party Certificate Authority (CA) and save it in the flash memory.

```
device(config)# ip ssl certificate-data-file tftp 10.10.10.1 cacert.pem
```

The following example shows how to change the key size for Ruckus-issued and imported digital certificates to 4096 bits.

```
device(config)# ip ssl cert-key-size 4096
```

The following example shows how to change the port number used for SSL communication.

```
device(config)# ip ssl port 334
```

The following example shows how to import an RSA private key from a client.

```
device(config)# ip ssl private-key-file tftp 192.168.9.210 keyfile
```

The following example shows how to configure the SSL certificate generation signing request for a country.

```
device(config)# ip ssl certificate country us
```

# ip ssl min-version

Configures the minimum TLS version to be used to establish the TLS connection.

## Syntax

**ip ssl min-version** { **tls_1_0** | **tls_1_1** | **tls_1_2** }

**no ip ssl min-version** { **tls_1_0** | **tls_1_1** | **tls_1_2** }

## Command Default

For devices which act as an SSL server or HTTPS server, the default connection is with TLS1.2.

For the Ruckus device which acts as the SSL client or the syslog, OpenFlow, or secure AAA client, the TLS version is decided based on the server support.

## Parameters

tls_1_0
: Specifies TLS 1.0 as the minimum version.

tls_1_1
: Specifies TLS 1.1 as the minimum version.

tls_1_2
: Specifies TLS 1.2 as the minimum version.

## Modes

Global configuration mode

## Usage Guidelines

If **tls_1_1** is set as the minimum version, TLS 1.1 and later versions are supported.

The **no** form of the command removes the minimum TLS version configuration and supports all TLS versions.

## Examples

The following example establishes the TLS connection using the TLS 1.1 version and above.

```
device(config)# ip ssl min-version tls_1_1
```

## History

| Release version | Command history |
|---|---|
| 08.0.20a | This command was introduced. |

# ip-subnet

Configures an IP subnet VLAN within a VLAN.

## Syntax

**ip-subnet** { *ip-address ip-mask* [ **name** *string* ] }

**no ip-subnet** { *ip-address ip-mask* [ **name** *string* ] }

## Command Default

A VLAN is not configured with an IP subnet and mask.

## Parameters

*ip-address*

Specifies the IP address you want to assign to a VLAN. The IP address can be in the format A.B.C.D or A.B.C.D/L, where L is the subnet mask length.

*ip-mask*

Specifies the subnet mask you want to assign. This is required when the subnet mask length is not specified along with the IP address.

**name** *string*

Specifies the name of the IP subnet. The name can be up to 32 characters in length.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the IP subnet VLAN.

## Examples

The following example shows how to configure an IP subnet VLAN within a VLAN.

```
device(config)# vlan 4
device(config-vlan-4)# ip-subnet 10.1.3.0/24 name Brown
```

# ip syslog source-interface

Configures an interface as the source IP address from which the syslog module sends log messages.

## Syntax

**ip syslog source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

**no ip syslog source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

## Command Default

When the management VRF is configured, the syslog module sends log messages only through ports that belong to the management VRF and through the out-of-band management port.

## Parameters

**ethernet** *stack-id/slot/port*
Specifies the Ethernet interface to be used as the source IP address.

**loopback** *number*
Specifies the loopback interface to be used as the source IP address.

**management** *number*
Specifies the management interface to be used as the source IP address.

**ve** *number*
Specifies the Virtual Ethernet interface to be used as the source IP address.

## Modes

Global configuration mode

## Usage Guidelines

When a source interface is configured, management applications use the lowest configured IP address of the specified interface as the source IP address in all the outgoing packets. If the configured interface is not part of the management VRF, the response packet does not reach the destination.

The syslog source interface configuration command **ip syslog source-interface** should be compatible with the management VRF configuration. Any change in the management VRF configuration takes effect immediately for syslog.

The **no** form of the command removes the configured interface as the source IP address.

## Examples

The following example configures an Ethernet interface as the source IP address for the syslog module to send log messages.

```
device(config)# ip syslog source-interface ethernet 1/1/1
```

The following example configures a management interface as the source IP address for the syslog module to send log messages.

```
device(config)# ip syslog source-interface management 1
```

# ip tacacs source-interface

Configures an interface as the source IP address from which the TACACS+ client establishes connections with TACACS+ servers.

## Syntax

**ip tacacs source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

**no ip tacacs source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

## Command Default

A TACACS+ source interface is not configured.

When a management VRF is configured, the TACACS+ client establishes connections with TACACS+ servers only through ports that belong to the management VRF and the out-of-band management port.

## Parameters

**ethernet** *stack-id/slot/port*
: Specifies the Ethernet interface to be used as the source IP address.

**loopback** *number*
: Specifies the loopback interface to be used as the source IP address.

**management** *number*
: Specifies the management interface to be used as the source IP address.

**ve** *number*
: Specifies the Virtual Ethernet interface to be used as the source IP address.

## Modes

Global configuration mode

## Usage Guidelines

For the TACACS+ client, a change in the management VRF configuration does not affect the existing TACACS+ connections. The changes are applied only to new TACACS+ connections.

The TACACS+ source interface configuration command **ip tacacs source-interface** must be compatible with the management VRF configuration.

The **no** form of the command removes the configured interface as the source IP address.

# Examples

The following example configures an Ethernet interface as the source IP address for the TACACS+ client to establish connections with TACACS+ servers.

```
device(config)# ip tacacs source-interface ethernet 1/1/1
```

The following example configures a Virtual Ethernet interface as the source IP address for the TACACS+ client to establish connections with TACACS+ servers.

```
device(config)# ip tacacs source-interface ve 1
```

# ip tcp burst-normal

Configures the threshold values for TCP SYN packets that are targeted at the router itself or passing through an interface.

## Syntax

**ip tcp burst-normal** *num-packets* **burst-max** *num-packets* **lockup** *time*

**no ip tcp burst-normal** *num-packets* **burst-max** *num-packets* **lockup** *time*

## Command Default

The threshold value is not configured.

## Parameters

*num-packets*
Configures the number of packets per second in normal burst mode. Valid values are from 1 through 100,000 packets per second.

**burst-max** *num-packets*
Specifies the number of packets per second in maximum burst mode. Valid values are from 1 through 100,000 packets per second.

**lockup** *time*
Configures the lockup period in seconds. Valid values are from 1 through 10,000 seconds.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

In a TCP SYN attack, an attacker floods a host with TCP SYN packets that have random source IP addresses. For each of these TCP SYN packets, the destination host responds with a SYN ACK packet and adds information to the connection queue. However, because the source host does not exist, no ACK packet is sent back to the destination host, and an entry remains in the connection queue until it ages out (after approximately one minute). If the attacker sends enough TCP SYN packets, the connection queue can fill up, and service can be denied to legitimate TCP connections.

To protect against TCP SYN attacks, you can configure the device to drop TCP SYN packets when excessive number of packets are encountered. You can set threshold values for TCP SYN packets that are targeted at the router itself or passing through an interface, and drop them when the thresholds are exceeded.

For Layer 3 router code, if the interface is part of a VLAN that has a router VE, you must configure TCP SYN attack protection at the VE level. When TCP SYN attack protection is configured at the VE level, it will apply to routed traffic only. It will not affect switched traffic.

**NOTE**
You must configure VLAN information for the port before configuring TCP SYN attack protection. You cannot change the VLAN configuration for a port on which TCP SYN attack protection is enabled.

**NOTE**
This command is available at the global configuration level on both chassis devices and compact devices. On chassis devices, this command is available at the interface level as well. This command is supported on Ethernet and Layer 3 interfaces.

The number of incoming TCP SYN packets per second is measured and compared to the threshold values as follows:

- If the number of TCP SYN packets exceeds the **burst-normal** value, the excess TCP SYN packets are dropped.
- If the number of TCP SYN packets exceeds the **burst-max** value, all TCP SYN packets are dropped for the number of seconds specified by the **lockup** value. When the lockup period expires, the packet counter is reset and measurement is restarted.

The **no** form of the command removes the threshold value set for TCP SYN packets.

# Examples

The following example sets the threshold value for TCP SYN packets targeted at the router.

```
device(config)# ip tcp burst-normal 10 burst-max 100 lockup 300
```

The following example sets the threshold value for TCP SYN packets received on interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip tcp burst-normal 10 burst-max 100 lockup 300
```

The following example sets the threshold value for TCP SYN packets received on VE 31.

```
device(config)# interface ve 31
device(config-vif-31)# ip tcp burst-normal 5000 burst-max 10000 lockup 300
```

# ip tcp keepalive

Configures the time interval between TCP keepalive messages.

## Syntax

**ip tcp keepalive** *timeout interval-time num-messages*

**no ip tcp keepalive** *timeout interval-time num-messages*

## Command Default

The time interval between TCP keepalive messages is not configured.

## Parameters

*timeout*

Configures the timeout in seconds to start sending keepalive messages. Set to 0 to disable the timeout.

*interval-time*

Configures the interval time in seconds between keepalive messages. Set to 0 to disable sending keepalive messages.

*num-messages*

Configures the number of keepalive messages to be sent before disconnecting.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables sending the keepalive messages. You can also set the *interval-time* variable as 0 to disable sending the keepalive messages.

## Examples

The following example configures the interval between TCP keepalive messages as 5 seconds.

```
device(config)# ip tcp keepalive 10 5 2
```

# ip-telephony data

Specifies the Avaya IP telephony data options in the DHCP server pool.

## Syntax

**ip-telephony data mcipadd** *ip-address* [ **tftpsrvr** *server-ip-address* | **httpsrvr** *server-ip-address* | **tlssrvr** *server-ip-address* | **mcport** *portnum* | **l2qaud** *prio* | **l2qsig** *prio* | **l2qvlan** *vlan-id* | **vlantest** *secs* ]

**no ip-telephony voice mcipadd** *ip-address* [ **tftpsrvr** *server-ip-address* | **httpsrvr** *server-ip-address* | **tlssrvr** *server-ip-address* | **mcport** *portnum* | **l2qaud** *prio* | **l2qsig** *prio* | **l2qvlan** *vlan-id* | **vlantest** *secs* ]

## Parameters

**mcipadd** *ip-address*
IP address of the gatekeeper. Atleast one IP address is required.

**mcport** *portnum*
Specifies IP telephony server port number. The default is 1719.

**tftpsrvr/httpsrvr/tlssrvr** *server ip-address*
Specifies the IP addresses of the TFP, HTTP, and TLS servers.

**l2qaud or l2qsig** *prio*
L2QAUD is the IP telephony L2 audio priority value. L2QSIG is the IP telephony L2 signaling priority value. This range is from 1 through 6. The default value is 6.

**l2qvlan** *vlan-id*
Specifies the IP telephony L2QVLAN number. The default is 0.

**vlantest** *secs*
The number of seconds a phone attempts to return to the previously known voice VLAN. This is not applicable for the default VLAN.

## Modes

DHCP server pool configuration mode

## Usage Guidelines

You must enter the MCIP address. The other parameters are optional.

The **no** form of the command removes the parameters from the DHCP server pool.

## Examples

The following example configures the MCIP address and MCPORT number for IP telephony data.

```
device(config)# ip dhcp-server pool cabo
device(config-dhcp-cabo)# ip-telephony data mcipadd 1.1.1.2 mcport 1719
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ip-telephony voice

Specifies the Avaya IP telephony voice options in the DHCP server pool.

## Syntax

**ip-telephony voice mcipadd** *ip-address* [ **tftpsrvr** *server-ip-address* | **httpsrvr** *server-ip-address* | **tlssrvr** *server-ip-address* | **mcport** *portnum* | **l2qaud** *prio* | **l2qsig** *prio* | **vlantest** *secs* ]

**no ip-telephony voice mcipadd** *ip-address* [ **tftpsrvr** *server-ip-address* | **httpsrvr** *server-ip-address* | **tlssrvr** *server-ip-address* | **mcport** *portnum* | **l2qaud** *prio* | **l2qsig** *prio* | **l2qvlan** *vlan-id* | **vlantest** *secs* ]

## Parameters

**mcipadd** *ip-address*
Specifies the addresses of gatekeepers. Atleast one IP address is required.

**mcport** *portnum*
Specifies the IP telephony server port number. The default is 1719.

**tftpsrvr/httpsrvr/tlssrvr** *server-ip-address*
Specifies the IP addresses of the TFTP, HTTP, and TLS servers.

**l2qaud or l2qsig** *prio*
Specifies the IP telephony L2QAUD or L2QSIG priority value. The range is from 1 to 6. The default value is 6.

**l2qvlan** *vlan-id*
Specifies the IP telephony L2QVLAN number. The default is 0.

**vlantest** *secs*
The number of seconds a phone attempts to return to the previously known voice VLAN. This is not applicable for the default VLAN.

## Modes

DHCP server pool configuration mode

## Usage Guidelines

You must enter the MCIP address. The other parameters are optional.

The **no** form of the command removes the parameters from the DHCP server pool.

## Examples

The following example configures the MCIP address and MCPORT number for IP telephony voice.

```
device(config)# ip dhcp-server pool cabo
device(config-dhcp-cabo)# ip-telephony voice mcipadd 1.1.1.2 mcport 1719
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ip telnet source-interface

Sets the lowest-numbered IP address configured on an interface as the device source for all Telnet packets.

## Syntax

**ip telnet source-interface** { **ethernet** *stack/slot/port* | **loopback** *loopback-num* | **management** *mgmt-num* | **ve** *ve-num* }

**no ip telnet source-interface** { **ethernet** *stack/slot/port* | **loopback** *loopback-num* | **management** *mgmt-num* | **ve** *ve-num* }

## Command Default

The default is the lowest-numbered IP address configured on the port through which the packet is sent.

## Parameters

**ethernet** *stack/slot/port*
: Configures the device source IP address for all Telnet packets as the IP address of the specified Ethernet interface.

**loopback** *loopback-num*
: Configures the device source IP address for all Telnet packets as the IP address of the specified loopback interface.

**mangement** *mgmt-num*
: Configures the device source IP address for all Telnet packets as the IP address of the specified management interface.

**ve** *ve-num*
: Configures the device source IP address for all Telnet packets as the IP address of the specified Virtual Ethernet (VE) interface.

## Modes

Global configuration mode

## Usage Guidelines

You can configure the Layer 3 switch to always use the lowest-numbered IP address on a specific Ethernet, loopback, or virtual interface as the source addresses for these packets. When configured, the Layer 3 switch uses the same IP address as the source for all packets of the specified type, regardless of the ports that actually sends the packets.

If your server is configured to accept packets only from specific IP addresses, you can use this configuration to simplify configuration of the server by configuring the device to always send the packets from the same link or source address.

If you specify a loopback interface as the single source for specified packets, servers can receive the packets regardless of the states of individual links. Thus, if a link to the server becomes unavailable but the client or server can be reached

through another link, the client or server still receives the packets, and the packets still have the source IP address of the loopback interface.

The **no** form of the command resets the source address of the packet as the lowest-numbered IP address on the interface that sends the packet.

# Examples

The following example configures the IP address of the Ethernet interface 1/1/1 as the source IP address for Telnet packets

```
device(config)# ip telnet source-interface ethernet 1/1/1
```

# ip tftp source-interface

Configures an interface as the source IP address from which TFTP sends and receives data and acknowledgments.

## Syntax

**ip tftp source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

**no ip tftp source-interface** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **management** *number* | **ve** *number* }

## Command Default

A TFTP source interface is not configured.

When a management VRF is configured, TFTP sends and receives data and acknowledgments only through ports that belong to the management VRF and through the out-of-band management port.

## Parameters

**ethernet** *stack-id/slot/port*
Specifies the Ethernet interface to be used as the source IP address.

**loopback** *number*
Specifies the loopback interface to be used as the source IP address.

**management** *number*
Specifies the management interface to be used as the source IP address.

**ve** *number*
Specifies the Virtual Ethernet interface to be used as the source IP address.

## Modes

Global configuration mode

## Usage Guidelines

Any change in the management VRF configuration takes effect immediately for TFTP. You cannot make changes in the management VRF configuration while TFTP is in progress.

The TFTP source interface configuration command **ip tftp source-interface** should be compatible with the management VRF configuration.

The **no** form of the command removes the configured interface as the source IP address.

## Examples

The following example configures an Ethernet interface as the source IP address for TFTP to send and receive data and acknowledgments.

```
device(config)# ip tftp source-interface ethernet 1/1/1
```

The following example configures a loopback interface as the source IP address for TFTP to send and receive data and acknowledgments.

```
device(config)# ip tftp source-interface loopback 1
```

# ip ttl

Modifies the time-to-live (TTL) threshold value.

## Syntax

**ip ttl** *threshold-value*

**no ip ttl** *threshold-value*

## Command Default

The default value for the TTL threshold is 64.

## Parameters

*threshold-value*

Sets the time TTL for packets on the network. The range is from 1 to 255 hops. The default is 64 hops.

## Modes

Global configuration mode

## Usage Guidelines

The time to live (TTL) threshold prevents routing loops by specifying the maximum number of router hops an IP packet originated by the Layer 3 switch can travel through. Each device capable of forwarding IP that receives the packet decrements (decreases) the packet TTL by one. If a device receives a packet with a TTL of 1 and reduces the TTL to zero, the device drops the packet.

The **no** form of the command resets TTL to 64 hops.

## Examples

The following example sets the TTL value to 25 hops.

```
device(config)# ip ttl 25
```

# ip use-acl-on-arp

Configures the ARP module to check the source IP address of the ARP request packets received on the interface before applying the specified ACL policies to the packet (ACL ARP filtering).

## Syntax

**ip use-acl-on-arp** [ *acl-num* ]

**no ip use-acl-on-arp** [ *acl-num* ]

## Command Default

ACL ARP filtering is not enabled.

## Parameters

*acl-num*

Specifies an ACL number to explicitly specify the ACL to be used for filtering.

## Modes

Interface configuration mode

## Usage Guidelines

ACL ARP filtering is not applicable to outbound traffic.

This command is available on devices running Layer 3 code. This filtering occurs on the management processor. The command is available on physical interfaces and virtual routing interfaces. ACLs used to filter ARP packets on a virtual routing interface can be inherited from a previous interface if the virtual routing interface is defined as a follower virtual routing interface. Only extended ACLs that use IP only can be used. If any other ACL is used, an error is displayed.

When the **ip use-acl-on-arp** command is configured, the ARP module checks the source IP address of the ARP request packets received on the interface. It then applies the specified ACL policies to the packet. Only the packet with the IP address that the ACL permits will be allowed to be written in the ARP table; those that are not permitted will be dropped.

ARP requests will not be filtered by ACLs if one of the following conditions occur:

- If the ACL is to be inherited from an IP ACL, but there is no IP ACL defined.
- An ACL ID is specified for the **ip use-ACL-on-arp** command, but no IP address or "any any" filtering criteria has been defined under the ACL ID.

The **no** form of the command disables the ACL ARP filtering.

# Examples

The following example shows a complete ACL ARP configuration.

```
device(config)# access-list 101 permit ip host 192.168.2.2 any
device(config)# access-list 102 permit ip host 192.168.2.3 any
device(config)# access-list 103 permit ip host 192.168.2.4 any
device(config)# vlan 2
device(config-vlan-2)# tag ethernet 1/1/1 to 1/1/2
device(config-vlan-2)# router-interface ve 2
device(config-vlan-2)# vlan 3
device(config-vlan-3)# tag ethernet 1/1/1 to 1/1/2
device(config-vlan-3)# router-int ve 3
device(config-vlan-3)# vlan 4
device(config-vlan-4)# tag ethernet 1/1/1 to 1/1/2
device(config-vlan-4)# router-int ve 4
device(config-vlan-4)# interface ve 2
device(config-ve-2)# ip access-group 101 in
device(config-ve-2)# ip address 192.168.2.1/24
device(config-ve-2)# ip use-acl-on-arp 103
device(config-ve-2)# exit
device(config)# interface ve 3
device(config-ve-3)# ip access-group 102 in
device(config-ve-3)# ip follow ve 2
device(config-ve-3)# ip use-acl-on-arp
device(config-ve-3)# exit
device(config-vlan-4)# interface ve 4
device(config-ve-4)# ip follow ve 2
device(config-ve-4)# ip use-acl-on-arp
device(config-ve-4)# exit
```

# ip vrrp auth-type

Configures the type of authentication used on a Virtual Router Redundancy Protocol (VRRP) interface.

## Syntax

**ip vrrp auth-type** { **no-auth** | **simple-text-auth** *auth-text* }

**no ip vrrp auth-type** { **no-auth** | **simple-text-auth** *auth-text* }

## Command Default

No authentication type is configured on a VRRP interface.

## Parameters

**no-auth**
Configures no authentication on the VRRP interface.

**simple-text-auth** *auth-text*
Configures a simple text string as a password used for authenticating packets on the interface. The maximum length of the text string is 64 characters.

## Modes

Interface configuration mode

## Usage Guidelines

If the **no-auth** option is configured, ensure that all interfaces on all devices that support the virtual router ID do not use authentication.

If the **simple-text-auth** option is configured, ensure that all interfaces on all devices that support the virtual router ID are configured to use simple password authentication with the same password.

The **no** form of this command removes the VRRP authentication from the interface.

**NOTE**
Authentication is not supported by VRRP-Ev3.

## Examples

The following example configures no authentication on Ethernet interface 1/1/6.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip vrrp auth-type no-auth
```

The following example configures simple password authentication on Ethernet interface 1/1/6.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip vrrp auth-type simple-text-auth yourpwd
```

# ip vrrp vrid

Configures an IPv4 Virtual Router Redundancy Protocol (VRRP) virtual router identifier (VRID).

## Syntax

**ip vrrp vrid** *vrid*

**no ip vrrp vrid** *vrid*

## Command Default

A VRRP VRID does not exist.

## Parameters

*vrid*

Configures a number for the IPv4 VRRP VRID. The range is from 1 through 255.

## Modes

Interface configuration mode

## Usage Guidelines

Before configuring this command, ensure that VRRP is enabled globally; otherwise, an error stating “Invalid input...” is displayed as you try to create a VRRP instance.

The **no** form of this command removes the IPv4 VRRP VRID from the configuration.

## Examples

The following example configures VRRP virtual router ID 1.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.5.1
device(config-if-e1000-1/1/6-vrid-1)# activate
VRRP router 1 for this interface is activating
```

# ip vrrp-extended auth-type

Configures the type of authentication used on a Virtual Router Redundancy Protocol Extended (VRRP-E) interface.

## Syntax

**ip vrrp-extended auth-type** { **no-auth** | **simple-text-auth** *auth-text* | **md5-auth** *auth-text* }

**no ip vrrp-extended auth-type** { **no-auth** | **simple-text-auth** *auth-text* | **md5-auth** *auth-text* }

## Command Default

No authentication is configured for a VRRP-E interface.

## Parameters

**no-auth**
Configures no authentication on the VRRP-E interface.

**simple-text-auth** *auth-text*
Configures a simple text string as a password used for authenticating packets on the interface. The maximum length of the text string is 64 characters.

**md5-auth** *auth-text*
Configures MD5 authentication on the interface. The maximum length of the text string is 64 characters.

## Modes

Interface configuration mode

## Usage Guidelines

If the **simple-text-auth** option is configured, ensure that all interfaces on all devices that support the virtual router ID are configured to use simple password authentication with the same password.

If the **md5-auth** option is configured, syslog and SNMP traps are generated if a packet is being dropped due to MD5 authentication failure. Using MD5 authentication implies that the software does not need to run checksum verification on the receiving device and can rely on the authentication code (message digest 5 algorithm) to verify the integrity of the VRRP-E message header.

Use the **show run** command with appropriate parameters to display the encrypted password; use the **enable password-display** command to display the unencrypted password.

If the **no-auth** option is configured, ensure that all interfaces on all devices that support the virtual router ID do not use authentication.

The **no** form of this command removes the VRRP-E authentication from the interface.

**NOTE**
Authentication is not supported by VRRP-Ev3.

# Examples

The following example configures no authentication on Ethernet interface 1/1/6.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip vrrp-extended auth-type no-auth
```

The following example configures simple password authentication on Ethernet interface 1/1/6.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip vrrp-extended auth-type simple-text-auth yourpwd
```

The following example configures MD5 authentication on Ethernet interface 1/1/6. When MD5 authentication is configured, a syslog message is displayed.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip vrrp-extended auth-type md5-auth lyk28d3j

Aug 10 18:17:39 VRRP: Configuration VRRP_CONFIG_MD5_AUTHENTICATION request received
Aug 10 18:17:39 VRRP: Port 1/1/6, VRID 2 - send advertisement
Ver:3 Type:1 Vrid:2 Pri:240 #IP:1 AuthType:2 Adv:1 Chksum:0x0000
HMAC-MD5 CODE:[00000000000000000000400010]
IpAddr: 10.53.5.1
```

# ip vrrp-extended vrid

Configures an IPv4 Virtual Router Redundancy Protocol Extended (VRRP-E) virtual router identifier (VRID).

## Syntax

**ip vrrp-extended vrid** *vrid*

**no ip vrrp-extended vrid** *vrid*

## Command Default

A VRRP-E VRID does not exist.

## Parameters

*vrid*

Configures a number for the IPv4 VRRP-E VRID. The range is from 1 through 255.

## Modes

Interface configuration mode

## Usage Guidelines

Before configuring this command, ensure that VRRP-E is enabled globally; otherwise an error stating "Invalid input..." is displayed as you try to create a VRRP-E instance.

The **no** form of this command removes the IPv4 VRRP-E VRID from the configuration.

## Examples

The following example configures VRRP-E VRID 1.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.10.1/24
device(config-if-e1000-1/1/6)# ip vrrp-extended vrid 1
device(config-if-e1000-1/1/6-vrid-1)# backup priority 50 track-priority 10
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.10.254
device(config-if-e1000-1/1/6-vrid-1)# activate
```

# ipsec profile

Creates an IP security (IPsec) profile and enters IPsec profile configuration mode.

## Syntax

**ipsec profile** *name*

**no ipsec profile** *name*

## Command Default

No IPsec profile is configured.

## Parameters

*name*

Specifies the name of an IPsec profile.

## Modes

Global configuration mode

## Usage Guidelines

An IPsec profile defines parameters for encrypting communications between IPsec peer devices.

After configuration, an IPsec profile is activated by attaching it to an IPsec virtual tunnel interface (VTI) by using the **tunnel protection ipsec profile** command in tunnel interface configuration mode.

The **no** form of the command removes the specified IPsec profile configuration.

## Examples

The following example shows how to create an IPsec profile named ipsec_profile and enters IPsec profile configuration mode for the profile.

```
device(config)# ipsec profile ipsec_profile
device(config-ipsec-profile-ipsec_profile)#
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ipsec proposal

Creates an IP security (IPsec) proposal and enters IPsec proposal configuration mode.

## Syntax

**ipsec proposal** *name*

## Parameters

*name*

Specifies the name of an IPsec proposal.

## Modes

Global configuration mode

## Usage Guidelines

An IPsec proposal defines an encryption algorithm, encapsulation mode, and transform set used to negotiate with a data path peer. An IPsec proposal is activated by attaching it to an IPsec profile.

There is a default IPsec proposal (**def-ipsec-prop**) that is defined at IPsec initialization and has the following settings:

- **transform**: ESP
- **encapsulation-mode**: Tunnel
- **encryption-algorithm**: AES-GCM-256

Use the **ipsec proposal** command to configure any additional IPsec proposals.

The **no** form of the command removes any IPsec proposal configuration other than the default IPsec proposal configuration.

The default IPsec proposal cannot be removed.

## Examples

The following example creates an IPSec proposal named ipsec_proposal and enters IPsec proposal configuration mode for the proposal .

```
device(config)# ipsec proposal ipsec_proposal
device(config-ipsec-proposal-ipsec_proposal)#
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# ipv6 access-list

Configures an IPv6 access control list (ACL) and enters IPv6 access list configuration mode.

## Syntax

**ipv6 access-list** *acl-name*

**no ipv6 access-list** *acl-name*

## Command Default

The IPv6 ACL is not configured.

## Parameters

*acl-name*

Specifies the ACL name, which must contain at least one alphabetic character.

## Modes

Global configuration mode

## Usage Guidelines

An ACL name must be unique among IPv6 and IPv4 ACLs.

The **no** form of the command removes the configured IPv6 ACL.

## Examples

The following example configures an IPv6 ACL named "acl1".

```
device(config)# ipv6 access-list acl1
device(config-ipv6-access-list acl1)#
```

# ipv6 address

Configures an IPv6 address for an interface.

## Syntax

**ipv6 address** *ipv6-prefix* [ **anycast** | **eui-64** ]

**no ipv6 address** *ipv6-prefix* [ **anycast** | **eui-64** ]

**ipv6 address** *ipv6-address* **link-local**

**no ipv6 address** *ipv6-address* **link-local**

## Command Default

An IPv6 address is not configured.

## Parameters

*ipv6-prefix*
Specifies the IPv6 prefix address in the format X:X::X:X/M.

**anycast**
Configuresan address as an anycast address.

**eui-64**
Configures the global address with an EUI-64 interface ID in the low-order 64 bits. The interface ID is automatically constructed in IEEE EUI-64 format using the interface's MAC address.

*ipv6-address*
Specifies the IPv6 address.

**link-local**
Configures the address as a link-local address.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command removes the IPv6 address.

## Examples

The following example configures an IPv6 address for the tunnel interface.

```
device(config)# interface tunnel 1
device(config-tnif-1)# ipv6 address 2001:DB8:384d:34::/64 eui-64
```

# ipv6 cache-lifetime

Configures the IPv6 cache-aging lifetime.

## Syntax

**ipv6 cache-lifetime** *interval*

**no ipv6 cache-lifetime** *interval*

## Command Default

Cache aging is enabled except on the Ruckus ICX 7750.

## Parameters

*interval*

Specifies the cache timeout interval in seconds. The default is 300 seconds.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of this command disables cache aging.

On the Ruckus ICX 7750, cache aging is disabled by default. You must ... to enable it.

## Examples

This example sets the cache-aging interval to 17 seconds.

```
Device (config)# ipv6 cache-lifetime 17
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was introduced. |

# ipv6 default-gateway

Configures the IPv6 address of the default gateway.

## Syntax

**ipv6 default-gateway** *ipv6-address*

**no ipv6 default-gateway**

## Parameters

*ipv6-address*
The IPv6 address of the default gateway.

## Modes

Global configuration mode

## Usage Guidelines

A device should have an IPv6 default gateway, for the following reasons:

- Although IPv6 discovers neighbors and routes dynamically, in some cases Router Advertisement (RA) and Router Solicitation (RS) operations are disabled and a default gateway is required to send traffic.
- Management devices (for example, TFTP servers, Telnet or SSH clients) are not members of the same subnet as the management IPv6 address.

If a management VLAN is not configured, the device can have only one IPv6 default gateway in the global configuration.

**NOTE**
If a management VLAN is configured (by means of the **default-ipv6-gateway** command in VLAN configuration mode), the device can have a maximum of 5 IPv6 default gateways with a metric (1 through 5) under the management VLAN.

Configured gateway addresses and the default gateway address must be in same subnet.

Use the **no** form of the command to remove the IPv6 address and disable the default gateway.

## Examples

The following example configures the IPv6 address of the IPv6 default gateway without a management VLAN configuration.

```
device# configure terminal
device(config)# ipv6 default-gateway 2001:DB8::/32
```

The following example removes and disables the IPv6 default gateway.

```
device(config)# no ipv6 default-gateway 2001:DB8::/32
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# Related Commands

default-ipv6-gateway

# ipv6 dhcp-relay destination

Enables the IPv6 DHCP relay agent function and specifies the IPv6 address as a destination address to which the client messages are forwarded.

## Syntax

**ipv6 dhcp-relay destination** *ipv6-address* [ **outgoing-interface** { **ethernet** *stack**/**slot**/**port* | **tunnel** *tunnel-id* | **ve** *ve-num* ]

**no ipv6 dhcp-relay destination** *ipv6-address* [ **outgoing-interface** { **ethernet** *stack**/**slot**/**port* | **tunnel** *tunnel-id* | **ve** *ve-num* ]

## Command Default

The IPv6 DHCP relay agent function is disabled.

## Parameters

*ipv6-address*
Specifies the IPv6 address as a destination address to which the client messages can be forwarded.

**outgoing-interface**
Configures the interface on which DHCPv6 packet will be relayed.

**ethernet** *stack**/**slot**/**port*
Specifies the Ethernet interface on which DHCPv6 packet will be relayed.

**tunnel** *tunnel-id*
Specifies the tunnel interface on which DHCPv6 packet will be relayed.

**ve** *ve-num*
Specifies the Virtual Ethernet (VE) interface on which DHCPv6 packet will be relayed.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command removes the DHCP relay agent from the interface.

You can configure up to 16 relay destination addresses on an interface.

# Examples

The following example enables the DHCPv6 relay agent function and specifies the relay destination (the DHCP server) address on an interface.

```
device(config)# interface ethernet 1/2/3
device(config-if-e10000-1/2/3)# ipv6 dhcp-relay destination 2001::2
device(config-if-e10000-1/2/3)# ipv6 dhcp-relay destination fe80::224:38ff:febb:e3c0 outgoing-interface
ethernet 1/2/5
```

# History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# ipv6 dhcp-relay distance

Assigns the administrative distance to IPv6 DHCP static routes installed in the IPv6 route table for the delegated prefixes on the interface.

## Syntax

**ipv6 dhcp-relay distance** *value*

**no ipv6 dhcp-relay distance** *value*

## Command Default

The administrative distance is not assigned.

## Parameters

*value*

Assigns the administrative distance to DHCPv6 static routes on the interface. The range is from 1 through 255. If the value is set to 255, then the delegated prefixes for this interface will not be installed in the IPv6 static route table.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command sets the parameter to a default value of 10.

The administrative distance value must be set so that it does not replace the same IPv6 static route configured by the user.

## Examples

The following example sets the administrative distance value to 25.

```
device(config-if-eth2/1)# ipv6 dhcp-relay distance 25
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# ipv6 dhcp-relay include-options

Includes the parameters on the IPv6 DHCP relay agent messages.

## Syntax

**ipv6 dhcp-relay include-options** [ **interface-id** ] [**remote-id** ] [ **link-layer-option** ]

**no ipv6 dhcp-relay include-options** [**interface-id** ] [**remote-id** ] [**link-layer-option** ]

## Command Default

The parameters are not included on the IPv6 DHCP relay agent messages.

## Parameters

**interface-id**
Includes the interface-ID parameter (option 18) in the IPv6 DHCP relay agent messages.

**remote-id**
Includes the remote-ID (option 37) parameter in the IPv6 DHCP relay agent messages.

**link-layer-option**
Includes the client link layer address (option 79) in the relay-forward messages.

## Modes

Interface configuration mode

## Usage Guidelines

The interface-ID parameter on the DHCPv6 relay forward message is used to identify the interface on which the client message is received. By default, this parameter is included only when the client message is received with the link-local source address.

You can enter either one or all of the include options as identifiers to specify in the relay-forward message.

The **no** form of the command disables the relay agent include options parameters.

## Examples

The following example includes the **link-layer-option** parameter on the DHCPv6 relay agent messages.

```
device(config)# interface ethernet 1/1/3
device(config-if-eth-1/1/3)# ipv6 dhcp-relay include-options link-layer-option
```

# History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced with support for the **interface-id** option. |
| 08.0.30 | Support for this command (with the **interface-id** keyword) was added in 08.0.30 and later releases. |
| 8.0.40 | Included support for **remote-id** and **link-layer-option** keywords. |

# ipv6 dhcp-relay maximum-delegated-prefixes

Sets the number of delegated prefixes that can be learned at the global and interface levels.

## Syntax

**ipv6 dhcp-relay maximum-delegated-prefixes** *value*

**no ipv6 dhcp-relay maximum-delegated-prefixes** *value*

## Command Default

The DHCPv6 Relay Agent Prefix Delegation Notification is enabled when the DHCPv6 relay agent feature is enabled on the interface.

## Parameters

*value*

Limits the maximum number of prefixes that can be learned at the global level. The range is from 0 through 512. The global level default value is 500 while the interface level default is 100.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of the command sets the parameter to the default value of the specified platform.

You can disable the DHCPv6 Relay Agent Prefix Delegation Notification at the system or the interface level by setting ipv6 dhcp-relay maximum-delegated prefixes to 0 at the system or interface level.

The sum of all the delegated prefixes that can be learned at the interface level is limited by the system maximum. Make sure that there is enough free space in the flash memory to save information about delegated prefixes in flash on both the active and standby management processors.

## Examples

The following example sets the maximum delegated prefixes to 500 at the global level.

```
device(config)# ipv6 dhcp-relay maximum-delegated-prefixes 500
```

The following example sets the maximum delegated prefixes to 100 at the interface level.

```
device(config)# config int e 1/2/1
device(config-if-e10000-1/2/1)# ipv6 dhcp-relay maximum-delegated-prefixes 100
```

# History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# ipv6 dhcp6 snooping vlan

Enables DHCPv6 snooping on a VLAN or a range of VLANs

## Syntax

**ipv6 dhcp6 snooping** *vlan-id* [ **to** *vlan-id* ... ]

**no ipv6 dhcp6 snooping** *vlan-id* [ **to** *vlan-id* ... ]

## Command Default

DHCPv6 snooping is disabled by default.

## Parameters

*vlan-id*

Specifies the ID of a configured client or DHCPv6 server VLAN.

**to** *vlan-id*

Specifies a range of VLANs.

## Modes

Global configuration mode

## Usage Guidelines

DHCPv6 snooping must be enabled on the client and the DHCPv6 server VLANs.

All VLANs included in the range when using the **to** keyword must be valid VLANs. Otherwise an error will occur.

The **no** form of the command disables DHCPv6 snooping on the VLAN.

## Examples

The following example enables DHCPv6 snooping on VLAN 2.

```
device# configure terminal
device(config)# ipv6 dhcp6 snooping vlan 2
```

The following example configures VLANs 100 through 150, VLAN 160, and VLANs 170 through 200 and enables DHCPv6 snooping on all of the configured VLANs.

```
device# configure terminal
device(config)# vlan 100 to 150
device(config-mvlan-100-150)# tagged ethernet 1/1/12
device(config-mvlan-100-150)# exit
device(config)# vlan 150 to 200
device(config-mvlan-150-200)# exit
device(config)# ipv6 dhcp6 snooping vlan 100 to 150 160 170 to 200
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | The **to** keyword was added to enable DHCPv6 snooping on a range of VLANs by using a single command. |

# ipv6 dns server-address

Configures IPv6 DNS server address.

## Syntax

**ipv6 dns server-address** *ipv6-address* [ *ipv6-address* ... ]

**no ipv6 dns server-address** *ipv6-address* [ *ipv6-address* ... ]

## Command Default

IPv6 DNS server addresses are not configured.

## Parameters

*ipv6-address*

Specifies the IPv6 address of the DNS server. You can specify up to four DNS server IPv6 address in the same command line.

## Modes

Global configuration mode

## Usage Guidelines

IPv6 defines new DNS record types to resolve queries for domain names to IPv6 addresses, as well as IPv6 addresses to domain names. Ruckus devices running IPv6 software support AAAA DNS records, which are defined in RFC 1886.

AAAA DNS records are analogous to the A DNS records used with IPv4. They store a complete IPv6 address in each record. AAAA records have a type value of 28.

The **no** form of the command removes the DNS server address.

## Examples

The following example configures an IPv6 DNS server address.

```
device(config)# ipv6 dns server-address 2001:DB8::1
```

# ipv6 enable

Enables IPv6.

## Syntax

**ipv6 enable**

**no ipv6 enable**

## Command Default

IPv6 is enabled by default in the Layer 2 switch code.

IPv6 is disabled by default in the router code.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

IPv6 is enabled by default in the Layer 2 switch code. If desired, you can disable IPv6 on a global basis on a device running the switch code.

IPv6 is disabled by default in the router code and must be configured on each interface that will support IPv6. In router code, the **ipv6 enable** command enables IPv6 on the switch and specifies that the interface is assigned an automatically computed link-local address

Before an IPv6 ACL can be applied to an interface, it must first be created, and then IPv6 must be enabled on that interface.

The **no** form of the command disables IPv6 on the interface.

## Examples

The following example re-enables the IPv6 after it has been disabled.

```
device(config)# ipv6 enable
```

The following example enables IPv6 on Ethernet interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ipv6 enable
```

# ipv6 hitless-route-purge-timer

Configures the timer to set the duration for which the routes should be preserved after switchover.

## Syntax

**ipv6 hitless-route-purge-timer** *seconds*

**no ipv6 hitless-route-purge-timer** *seconds*

## Command Default

The default timer setting is 45 seconds.

## Parameters

*seconds*

Specifies the time after switchover to start IPv6 route purge. The value can range from 2 to 600 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the configured value and sets the timer to the default 45 seconds.

## Examples

The following example shows how to set the IPv6 hitless purge timer to 75 seconds.

```
device(config)# ipv6 hitless-route-purge-timer 60
```

# ipv6 hop-limit

Configures the maximum number of hops an IPv6 packet can traverse.

## Syntax

**ipv6 hop-limit** *number*

**no ipv6 hop-limit** *number*

## Command Default

By default, the maximum number of hops an IPv6 packet can traverse is 64.

## Parameters

*number*

Specifies the maximum number of hops an IPv6 packet can traverse. Valid values are 0 through 255. The default value is 64.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the maximum number of hops an IPv6 packet can traverse to 64.

## Examples

The following example configures the maximum number of hops an IPv6 packet can traverse to 70.

```
device(config)# ipv6 hop-limit 70
```

# ipv6 icmp error-interval

Configures ICMP rate limiting, that is, the rate at which IPv6 ICMP error messages are sent out on a network.

## Syntax

**ipv6 icmp error-interval** *interval* [ *size* ]

**no ipv6 icmp error-interval** *interval* [ *size* ]

## Command Default

ICMP rate limiting is enabled by default.

## Parameters

*interval*

Specifies the interval in milliseconds at which tokens are placed in the bucket. Valid values are 0 through 2147483647 and the default value is 100 milliseconds. Setting the value to 0 disables ICMP rate limiting.

*size*

Specifies the maximum number of tokens stored in the bucket. Valid values are1 to 200 and the default is 10 tokens.

## Modes

Global configuration mode

## Usage Guidelines

You can limit the rate at which IPv6 ICMP error messages are sent out on a network. IPv6 ICMP implements a token bucket algorithm.

To illustrate how this algorithm works, imagine a virtual bucket that contains a number of tokens. Each token represents the ability to send one ICMP error message. Tokens are placed in the bucket at a specified interval until the maximum number of tokens allowed in the bucket is reached. For each error message that ICMP sends, a token is removed from the bucket. If ICMP generates a series of error messages, messages can be sent until the bucket is empty. If the bucket is empty of tokens, error messages cannot be sent until a new token is placed in the bucket.

If you configure the interval value to a number that does not evenly divide into 100000 (100 milliseconds), the system rounds up the value to a next higher value that does divide evenly into 100000. For example, if you specify an interval value of 150, the system rounds up the value to 200.

The **no** form of the command disables ICMP rate limiting.

## Examples

The following example configures the interval to 1000 milliseconds and the number of tokens to 100 tokens.

```
device(config)# ipv6 icmp error-interval 1000 100
```

# ipv6 icmp fragment_header_bit

Sets the atomic fragment header bit when the maximum transmission unit (MTU) is less than or equal to 1280.

## Syntax

**ipv6 icmp fragment_header_bit**

**no ipv6 fragment_header_bit**

## Command Default

Disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

Use this command only if USG IPV6 functionality is needed.

The **no** form of the command restores the default.

## Examples

The following example configures the fragment header bit.

```
device# configure terminal
device(config)# ipv6 icmp fragment_header_bit
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# ipv6 icmp source-route

Generates ICMP parameter problem message for source routed IPv6 packet.

## Syntax

**ipv6 icmp source-route**

**no ipv6 icmp source-route**

## Command Default

By default, when the router drops a source-routed packet, it sends an ICMP Parameter Problem (type 4), Header Error (code 0) message to the packet's source address, pointing to the unrecognized routing type.

## Modes

Global configuration mode

## Usage Guidelines

The IPv6 specification (RFC 2460) specifies support for IPv6 source-routed packets using a type 0 Routing extension header, requiring device and host to process the type 0 routing extension header. However, this requirement may leave a network open to a DoS attack. A security enhancement disables sending IPv6 source-routed packets to IPv6 devices. (This enhancement conforms to RFC 5095.)

By default, when the router drops a source-routed packet, it sends an ICMP Parameter Problem (type 4), Header Error (code 0) message to the packet's source address, pointing to the unrecognized routing type.

The **no** form of the command disables the ICMP error messages for source routed IPv6 packet.

## Examples

The following example disables the ICMP error messages for source routed IPv6 packet.

```
device(config)# no ipv6 icmp source-route
```

The following example re-enables the ICMP error messages for source routed IPv6 packet.

```
device(config)# ipv6 icmp source-route
```

# ipv6 load-sharing

Enables Equal-cost multi-path routing (ECMP) load sharing for IPv6.

## Syntax

**ipv6 load-sharing** [ *num* ]

**no ipv6 load-sharing** [ *num* ]

## Command Default

ECMP load-sharing for IPv6 is enabled and allows traffic to be balanced across up to four equal paths.

## Parameters

*num*

Specifies the number of load-sharing paths. The value can range from 2 through 8. The default value is 4.

## Modes

Global configuration mode.

## Usage Guidelines

If you want to re-enable the feature after disabling it, you must specify the number of load-sharing paths.

The **no** form of the command sets the load-sharing path to the default value of 4.

## Examples

The following example sets the number of ECMP load-sharing paths for IPv6 to 6.

```
device(config)# ipv6 load-sharing 6
```

# ipv6 max-mroute

Configures the maximum number of IPv6 multicast routes that are supported.

## Syntax

**ipv6 max-mroute** *num*

**no ipv6 max-mroute** *num*

## Command Default

No maximum number of supported routes is configured.

## Parameters

*num*

Configures the maximum number of multicast routes supported.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the default (no maximum number of supported routes is configured).

## Examples

The following example configures the maximum number of 20 supported IPv6 multicast routes on the VRF named my_vrf.

```
Device(config)# vrf my_vrf
Device(config)# address-family ipv6
Device(config-vrf)# ipv6 max-mroute 20
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ipv6 mld group-membership-time

Specifies the multicast listener discovery (MLD) group membership time for the default VRF or for a specified VRF.

## Syntax

**ipv6 mld group-membership-time** *num*

**no ipv6 mld group-membership-time** *num*

## Command Default

An MLD group will remain active on an interface in the absence of a group report for 260 seconds, by default.

## Parameters

*num*

Number in seconds, from 5 through 26000.

## Modes

Global configuration mode.

VRF configuration mode.

## Usage Guidelines

The **no** form of this command resets the group membership time interval to the default of 260 seconds.

Group membership time defines how long a group will remain active on an interface in the absence of a group report.

## Examples

This example specifies an MLD group membership time of 2000 seconds for the default VRF.

```
device# configure terminal
device(config)# ipv6 mld group-membership-time 2000
```

This example specifies an MLD group membership time of 2000 seconds for a specified VRF.

```
device# configure terminal
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# ipv6 mld group-membership-time 2000
```

# ipv6 mld llqi

Configures the multicast listener discovery (MLD) last listener query interval.

## Syntax

**ipv6 mld llqi** *seconds*

**no ipv6 mld llqi** *seconds*

## Command Default

The MLD last listener query interval is 1 second.

## Parameters

*seconds*

specifies the number in seconds, of MLD group addresses available for all VRFs. The range is 1 through 25; the default is 1.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the default MLD last listener query interval.

Any MLD group memberships exceeding the group limit are not processed.

The last listener query interval is the maximum response delay inserted into multicast address-specific queries sent in response to Done messages, and is also the amount of time between multicast address-specific query messages. When a device receives an MLD Version 1 leave message or an MLD Version 2 state-change report, it sends out a query and expects a response within the time specified by the last listener query interval. Configuring a lower value for the last listener query interval allows members to leave groups faster.

## Examples

This example configures a last listener query interval of 5 seconds.

```
Device(config)# ipv6 mld llqi 5
```

This example configures a last listener query interval of 5 seconds for a VRF.

```
Device(config)# ipv6 router pim vrf blue
Device(config-ipv6-pim-router-vrf-blue)# ipv6 mld llqi 5
```

# ipv6 mld max-group-address

Configures the maximum number of MLD group addresses for VRFs.

## Syntax

**ipv6 mld max-group-address** *num*

**no ipv6 mld max-group-address** *num*

## Command Default

The default value is 4096.

## Parameters

*num*

Specifies the maximum number of MLD group addresses available, either for the default VRF or for the specified VRF. The range is 1 through 8192.

## Modes

Global configuration mode

VRF configuration sub-mode

## Usage Guidelines

If the **no** form of this command is configured, the maximum number of MLD group addresses is reset to the default.

Any MLD group memberships exceeding the group limit are not processed.

## Examples

The following example configures a maximum of 1000 MLD group addresses for the default VRF.

```
device# configure terminal
device(config)# ipv6 mld max-group-address 1000
```

The following example configures a maximum of 1000 MLD group addresses for the VRF named blue.

```
device# configure terminal
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# ipv6 mld max-group-address 1000
```

# ipv6 mld max-response-time

Configures the maximum time a multicast listener has to respond to queries for the default virtual routing and forwarding (VRF) instance or for a specified VRF.

## Syntax

**ipv6 mld max-response-time** *num*

**no ipv6 mld max-response-time** *num*

## Command Default

If this command is not configured, the maximum time a multicast listener has to respond to queries is 10 seconds.

## Parameters

*num*

specifies the maximum time, in seconds, a multicast listener has to respond. The range is 1 through 25; the default is 10.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

If the **no** form of this command is configured, the maximum time a multicast listener has to respond to queries is 10 seconds.

## Examples

The following example configures the maximum time a multicast listener has to respond to queries to 20 seconds.

```
device# configure terminal
device(config)# ipv6 mld max-response-time 20
```

The following example configures the maximum time a multicast listener has to respond to queries to 20 seconds for the VRF named vpn1.

```
device# configure terminal
device(config)# vrf vpn1
Device(config-vrf-vpn1)# address-family ipv6
device(config)# ipv6 mld max-response-time 20
```

# ipv6 mld port-version

Configures the multicast listening discovery (MLD) version on a virtual Ethernet interface.

## Syntax

**ipv6 mld port-version** *version-number*

**no ipv6 mld port-version**

## Command Default

The port uses the MLD version configured globally.

## Parameters

*version-number*
Specifies the MLD version, 1 or 2.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command restores the MLD version configured globally.

## Examples

This example configures MLD version 2 on virtual Ethernet interface 10.

```
device# configure terminal
device(config)# interface ve 10
device(config-vif-10)# ipv6 mld port-version 2
```

# ipv6 mld query-interval

Configures the frequency at which multicast listening discovery (MLD) query messages are sent.

## Syntax

**ipv6 mld query-interval** *num*

**no ipv6 mld query-interval** *num*

## Command Default

125 seconds

## Parameters

*num*

Number in seconds, from 2 through 3600. The default is 125.

## Modes

Global configuration mode.

VRF configuration mode.

## Usage Guidelines

The **no** form of this command resets the query interval to the default of 125 seconds.

You must specify a query-interval value that is greater than the interval configured by the `ipv6 mld max-response-time` command.

## Examples

This example sets the MLD query interval to 50 seconds.

```
Device(config)# ipv6 mld query-interval 50
```

This example sets the MLD query interval for a VRF to 50 seconds.

```
Device(config)# ipv6 router pim vrf blue
Device(config-ipv6-pim-router-vrf-blue)# ipv6 mld query-interval 50
```

# ipv6 mld robustness

Configures the number of times that the device sends each multicast listening discovery (MLD) message from an interface.

## Syntax

**ipv6 mld robustness** *num*

**no ipv6 mld robustness** *num*

## Command Default

The MLD robustness is 2 seconds.

## Parameters

*num*

Number in seconds, from 2 through 7. The default is 2.

## Modes

Global configuration mode.

VRF configuration mode.

## Usage Guidelines

The **no** form of this command resets the query interval to the default of 2 seconds.

Configure a higher value to ensure high MLD reliability.

## Examples

This example configures the MLD robustness to 3 seconds.

```
Device(config)# ipv6 mld robustness 3
```

This example configures the MLD robustness for a VRF to 3 seconds.

```
Device(config)# ipv6 router pim vrf blue
Device(config-ipv6-pim-router-vrf-blue)# ipv6 mld robustness 3
```

# ipv6 mld static-group

Configures one or more physical ports to be a permanent (static) member of a multicast listening discovery (MLD) group based on the range or count.

## Syntax

**ipv6 mld static-group** *multicast-group-addr* [ **count** *count-number* | **to** *multicast-group-addr* ] [ **ethernet** *stackid*/*slot*/*portnum* ] [ **ethernet** *stackid*/*slot*/*portnum* **to ethernet** *stackid*/*slot*/*portnum* ] ]

**no ipv6 mld static-group** *multicast-group-addr* [ **count** *count-number* | **to** *multicast-group-addr* ] [ **ethernet** *stackid*/*slot*/*portnum* ] [ **ethernet** *stackid*/*slot*/*portnum* **to ethernet** *stackid*/*slot*/*portnum* ] ]

## Command Default

The port is not added to MLD group.

## Parameters

*ip-addr*

The address of the static MLD group.

**count** *count-number*

Specifies the number of static MLD groups The range is 2 through 256.

**to**

Specifies a range of addresses.

**ethernet** *stackid*/*slot*/*portnum*

Specifies the ID of the physical port that will be a member of the MLD group. On standalone devices specify the interface ID in the format slot/port-id; on stacked devices you must also specify the stack ID, in the format stack-id/slot/port-id. You can configure a single port or a list of ports, separated by a space.

## Modes

Interface configuration mode.

## Usage Guidelines

The **no** form of this command removes the port or ports from the MLD group.

You can specify as many port numbers as you want to include in the static group.

For a virtual routing interface (ve), specify the physical Ethernet ports on which to add the group address.

# Examples

The following example configures two static groups, starting from ff0d::1, without having to receive an MLDv1 report on a virtual Ethernet interface,

```
device# configure terminal
device(config)# interface ethernet 10000 1/1/2
device(config-if-e10000-1/1/2)# ipv6 mld static-group ff0d::1 count 2
```

The following example configures two static MLD groups, starting from ff0d::1, using the **to** keyword.

```
device# configure terminal
device(config)# interface ethernet 10000 1/1/2
device(config-if-e10000-1/1/2)# ipv6 mld static-group ff0d::1 to ff0d::2
```

The following example configures two static MLD groups on virtual ports starting from ff0d::1 using the **count** keyword.

```
device# configure terminal
device(config)# interface ve 10
device(config-vif-10)# ipv6 mld static-group ff0d::1 count 2 ethernet 1/5/2
```

The following example configures two static groups on virtual ports starting from ff0d::1 using the **to** keyword.

```
device# configure terminal
device(config)# interface ve 10
device(config-vif-10)# ipv6 mld static-group ff0d::1 to ff0d::2 ethernet 1/5/2
```

# ipv6 mld tracking

Enables multicast listening discovery (MLD) tracking on a virtual interface.

## Syntax

**ipv6 mld tracking**

**no ipv6 mld tracking**

## Command Default

Multicast tracking is disabled on the virtual interface.

## Modes

Virtual interface configuration mode

## Usage Guidelines

The **no** form of this command restores the default; tracking is disabled.

When MLD tracking is enabled, a Layer 3 device tracks all clients that send membership reports. When a Leave message is received from the last client, the device immediately stops forwarding to the physical port, without waiting 3 seconds to confirm that no other clients still want the traffic.

## Examples

This example enables multicast tracking on a virtual interface.

```
device# configure terminal
device(config)# interface ve 13
device(config-vif-13)# ipv6 mld tracking
```

# ipv6 mld version

Configures the multicast listening discovery (MLD) version for snooping on an interface.

## Syntax

**ipv6 mld version** { **1** | **2** }

**no ipv6 mld version** { **1** | **2** }

## Command Default

MLD Version 1 is configured.

## Parameters

**1**

Configures MLD version 1.

**2**

Configures MLD version 2.

## Modes

Global configuration mode

Interface configuration mode

IPv6 PIM router configuration mode

## Usage Guidelines

The default MLD version when PIM Sparse Mode (PIM-SM) is enabled on an interface is MLDv1. You must configure the version 2 to enable MLDv2.

The **no** form of this command restores the default, that is, 1.

## Examples

The following example configures MLD version 2 globally.

```
device(config)# ipv6 mld version 2
```

The following example configures MLD version 2 for a specified VRF.

```
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# ipv6 mld version 2
```

The following example configures MLD version 2 on an interface.

```
device(config)# interface ve 10
device(config-vif-10)# ipv6 mld version 2
```

The following example enables MLDv2.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# ipv6 mld version 2
```

# ipv6 mroute

Configures a static IPv6 route to direct multicast traffic along a specific path.

## Syntax

**ipv6 mroute** [ **vrf** *vrf-name* ] *ipv6-address-prefix/prefix-length* { **ethernet** *unit* **/** *slot* **/** *port* | **ve** *num* | **tunnel** *num* } [*cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

**no ipv6 mroute** [ **vrf** *vrf-name* ] *ipv6-address-prefix/prefix-length* { **ethernet** *unit* **/** *slot* **/** *port* | **ve** *num* | **tunnel** *num* } [*cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

## Command Default

No static IPv6 multicast route is configured.

## Parameters

**vrf** *vrf-name*

Configures a static mroute for this virtual routing and forwarding (VRF) route.

*ipv6-address-prefix/prefix-length*

Configures the destination IPv6 address and prefix for which the route should be added.

**ethernet** *unit* **/** *slot* **/***port*

Configures an Ethernet interface as the route path.

**ve** *num*

Configures a virtual interface as the route path.

**tunnel** *num*

Configures a tunnel interface as the route path.

*cost*

Configures a metric for comparing the route to other static routes in the IPv6 static route table that have the same destination. The range is 1 to 16; the default is 1.

**distance** *distance-value*

Configures the route's administrative distance. The range is 1 to 255. The default is 1.

**name** *name*

Name for this static route.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

The **no** form of this command deletes a previously configured static multicast route.

The **ethernet** *unit/slot/port* designation for the destination does not apply to PIM SM.

Connected routes on PIM-enabled interfaces are automatically added to the mRTM table.

# Examples

The following example configures a static IPv6 mroute to directly connected network 2020::0/120 on virtual interface ve 130.

```
Device# configure terminal
Device(config)# ipv6 mroute 2020::0/120 ve 130
```

The following example configures a static IPv6 mroute within a VRF called vpn1. The VRF has a route descriptor of 100:200. IPv6 addressing is specified for the VRF. The static multicast route has a destination of 2001:0DB8:0:1::1/120, and the address of the next hop gateway is 5100::192:1:1:1.

```
Device# configure terminal
Device(config)# ipv6 unicast-routing
Device(config)# vrf vpn1
Device (config-vrf-vpn1)# rd 100:200
Device (config-vrf-vpn1)# address-family ipv6
Device (config-vrf-vpn1-ipv6)# ipv6 mroute 2001:0DB8:0:1::1/120 5100::192:1:1:1
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ipv6 mroute (next hop)

Configures a static IPv6 multicast route (mroute) with a next hop.

## Syntax

**ipv6 mroute** [ **vrf** *vrf-name* ] *ipv6-address-prefix/prefix-length next-hop address* [ *cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

**no ipv6 mroute** [ **vrf** *vrf-name* ] *ipv6-address-prefix/prefix-length next-hop address* [ *cost* ] [ **distance** *distance-value* ] [ **name** *name* ]

## Command Default

No next-hop static IPv6 multicast route is configured.

## Parameters

**vrf** *vrf-name*
Configures a static mroute for this virtual routing and forwarding (VRF) route.

*ipv6-address-prefix/prefix-length*
Configures the destination IPv6 address and prefix for which the route should be added.

*next-hop address*
Configures a next-hop address as the route path.

*cost*
Configures a metric for comparing the route to other static routes in the static route table that have the same destination. The range is 1-16; the default is 1.

**distance** *distance-value*
Configures the route's administrative distance. The range is 1 to 255; the default is 1.

**name** *name*
Name for this static route.

## Modes

VRF configuration mode

## Usage Guidelines

The **no** form of this command deletes a previously configured next-hop static IPv6 multicast route.

## Examples

The following example configures a next-hop static multicast IPv6 route to network 2020::0/120 with 2022::0/120 as the next hop.

```
Device(config-vrf)# ipv6 mroute 2020::0/120 2022::0/120
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ipv6 mroute next-hop-enable-default

Enables the option to use the default multicast route (mroute) to resolve a static IPv6 mroute next hop.

## Syntax

**ipv6 mroute** [ **vrf** *vrf-name* ] **next-hop-enable-default**

**no ipv6 mroute** [ **vrf** *vrf-name* ] **next-hop-enable-default**

## Command Default

IPv6 multicast static routes are not resolved using the default multicast static route.

## Parameters

**vrf** *vrf-name*

Configures a static mroute for the specified virtual routing and forwarding (VRF) route.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

Before configuring an IPv6 multicast static route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and must enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

The **no** form of the command disables IPv6 static route next-hop resolution through the default route. If a VRF is configured, the **no** form of the command removes the static IPv6 route configuration from the VRF.

## Examples

The following example configures static routing next-hop recursion to three levels (the default). It configures the network default static route and allows it to resolve other static routes.

**NOTE**
You can specify a level of recursion up to 10.

```
device# configure terminal
device(config)# ipv6 mroute next-hop-recursion
device(config)# ipv6 mroute 0.0.0.0 0.0.0.0 xxx.xxx.xxx.xxx
device(config)# ipv6 mroute next-hop-enable-default
```

The following example enables the VRF named vpn1 to resolve an IPv6 multicast static route through the default IPv6 multicast static route, after configuring IPv6 on the device, setting a route descriptor for the VPN, and specifying IPv6 addressing be used on the VPN.

```
Device# configure terminal
Device(config)# ipv6 unicast-routing
Device(config)# vrf vpn1
Device (config-vrf-vpn1)# rd 100:200
Device (config-vrf-vpn1)# address-family ipv6
Device(config-vrf)# ipv6 mroute next-hop-enable-default
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ipv6 mroute next-hop-recursion

Configures the recursion level for resolving an IPv6 multicast static route static.

## Syntax

**ipv6 mroute** [ **vrf** *vrf-name* ] **next-hop-recursion** [ *number* ]

**no ipv6 mroute** [ **vrf** *vrf-name* ] **next-hop-recursion** [ *number* ]

## Command Default

By default, only the local IPv6 address table is consulted to resolve the next hop toward a multicast static route destination.

## Parameters

**vrf** *vrf-name*

Specifies the VRF that contains the next-hop router (gateway) for the route.

*number*

Specifies the level of recursion for address lookup. The range is 1 through 10. If no number is specified, the default value is 3.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and must enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

The **no** form of the command disables IPv6 multicast static route next-hop recursion. If a VRF is configured, the **no** form of the command removes the static IPv6 route configuration from the VRF.

## Examples

The following example configures recursive IPv6 multicast static route lookup to five levels.

```
device# configure terminal
device(config)# ipv6 mroute next-hop-recursion 5
```

The following example configures recursive lookup to seven levels for the VRF named vpn2. The VRF has a route descriptor of 100:200. IPv6 addressing is specified for the VRF.

```
Device# configure terminal
Device(config)# ipv6 unicast-routing
Device(config)# vrf vpn2
Device (config-vrf-vpn2)# rd 100:200
Device (config-vrf-vpn2)# address-family ipv6
Device (config-vrf-vpn2-ipv6)# ipv6 mroute next-hop-recursion 7
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# ipv6 mtu

Configures the IPv6 MTU on individual interfaces.

## Syntax

**ipv6 mtu** *unit*

**no ipv6 mtu** *unit*

## Command Default

By default, in non-jumbo mode, the default and maximum Ethernet MTU size is 1500 bytes. When jumbo mode is enabled, the default Ethernet MTU size is 9216.

## Parameters

*unit*

Specifies the maximum length of an IPv6 packet that can betransmitted on a particular interface. Valid values are between 1280 and 1500, or 1280 and 10182 if jumbo mode is enabled.

## Modes

Interface configuration mode

## Usage Guidelines

The IPv6 maximum transmission unit (MTU) is the maximum length of an IPv6 packet that can be transmitted on a particular interface. If an IPv6 packet is longer than an MTU, the host that originated the packet fragments the packet and transmits its contents in multiple packets that are shorter than the configured MTU.

By default, in non-jumbo mode, the default and maximum Ethernet MTU size is 1500 bytes. When jumbo mode is enabled, the default Ethernet MTU size is 9216. The maximum Ethernet MTU size is 10218.

The IPv6 MTU functionality is applicable to VEs and physical IP interfaces. It applies to traffic routed between networks. The minimum IPv4 and IPv6 MTU values for both physical and virtual interfaces are 1280.

IPv6 MTU cannot be configured globally. It is supported only on devices running Layer 3 software.

The **no** form of the command resets the MTU to the default values.

## Examples

The following example configures the MTU on Ethernet interface 1/3/1 as 1280 bytes.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 mtu 1280
```

# ipv6 multicast

Globally sets the multicast listening discovery (MLD) snooping mode to active.

## Syntax

**ipv6 multicast** [ **active** | **passive** ]

**no ipv6 multicast** [ **active** | **passive** ]

## Command Default

MLD mode is passive.

## Parameters

**active**

Specifies that the device actively sends out MLD queries to identify IPv6 multicast groups on the network, and makes entries in the MLD table based on the group membership reports it receives from the network.

**passive**

Specifies that the device forwards reports to the router ports which receive queries. MLD snooping in passive mode does not send queries, but does forward queries to the entire VLAN.

## Modes

Global configuration mode

## Usage Guidelines

If you specify an MLD mode for a VLAN, the MLD mode overrides the global setting.

In active MLD mode, a device actively sends out MLD queries to identify IPv6 multicast groups on the network, and makes entries in the MLD table based on the group membership reports it receives from the network. In passive MLD mode, the device forwards reports to the router ports that receive queries. MLD snooping in passive mode does not send queries, but does forward queries to the entire VLAN.

**NOTE**
The **ipv6 multicast** command replaces the **ipv6 mld-snooping** command. The **multicast6** command replaces the **mld-snooping** command.

The **no** form of this command when the **active** parameter is used stops the device from sending out MLD queries to identify IPv6 multicast groups on the network. The **no** form of the command when used with the **passive** parameter stops forwarding reports to the router ports which receive queries.

## Examples

The following example globally sets the MLD snooping mode to active.

```
device(config)# ipv6 multicast active
```

# ipv6 multicast age-interval

Configures the time that group entries can remain in a multicast listening discovery (MLD) group table.

## Syntax

**ipv6 multicast age-interval** *interval*

**no ipv6 multicast age-interval** *interval*

## Command Default

Group entries can remain in the MLD group table for up to 260 seconds.

## Parameters

*interval*

Specifies the time, in seconds, that group entries can remain in the MLD group table. The range is 20 through 7200 seconds. The default is 260 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default age interval to 260 seconds.

When a device receives a group membership report it makes an entry for that group in the MLD group table. You can configure the **ipv6 multicast age-interval** to specify how long the entry can remain in the table before the device receives another group membership report. When multiple devices are connected, they must all be configured for the same age interval, which must be at least twice the length of the query interval, so that missing one report does not stop traffic.

Non-querier age intervals must be the same as the age interval of the querier.

## Examples

This example configures the MLD group-table age interval to 280 seconds.

```
Device(config)#ipv6 multicast age-interval 280
```

# ipv6 multicast disable-flooding

Disables the flooding of unregistered IPv6 multicast frames in an MLD-snooping-enabled VLAN.

## Syntax

**ipv6 multicast disable-flooding**

**no ipv6 multicast disable-flooding**

## Command Default

The device floods unregistered IPv6 multicast frames in an MLD-snooping-enabled VLAN.

## Modes

Global configuration mode

## Usage Guidelines

**NOTE**
Disabling the flooding of unregistered IPv6 multicast frames in an MLD-snooping-enabled VLAN is supported only on the ICX 7750 (standalone and stacking) platform.

The **no** form of this command enables the flooding of unregistered IPv6 multicast frames in an MLD-snooping-enabled VLAN.

In releases prior to FastIron 8.0.30, support for this command on the Ruckus ICX 7750 was for devices in standalone mode only.

After the hardware forwarding database (FDB) entry is made, the multicast traffic is switched only to the VLAN hosts that are members of the multicast group. This can avoid congestion and loss of traffic on the ports that have not subscribed to this IPv6 multicast traffic.

## Examples

The following example disables flooding of unregistered IPv6 multicast frames.

```
device(config)# ipv6 multicast disable-flooding
```

## History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced. |

# ipv6 multicast leave-wait-time

Configures the wait time before stopping traffic to a port when a leave message is received.

## Syntax

**ipv6 multicast leave-wait-time** *num*

**no ipv6 multicast leave-wait-time** *num*

## Command Default

The wait time is 2 seconds.

## Parameters

*num*

Specifies the time, in seconds, the device should wait before stopping traffic to a port when a leave message is received The range is 1 through 5 seconds. The default is 2 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default wait time.

The device sends group-specific queries once per second to ask if any client in the same port still needs the group. Because of internal timer granularity, the actual wait time is between n and (n+1) seconds (n is the configured value).

## Examples

This example configures the maximum time a client can wait before responding to a query as 1 second.

```
Device(config)#ipv6 multicast leave-wait-time 1
```

# ipv6 multicast max-response-time

Sets the maximum number of seconds a client (IPv6) can wait before responding to a query sent by the device.

## Syntax

**ipv6 multicast max-response-time** *interval*

**no ipv6 multicast max-response-time** *interval*

## Command Default

The wait time is 10 seconds.

## Parameters

*interval*

Specifies the maximum time, in seconds, a client can wait before responding to a query sent by the switch. The range is 1 through 25 seconds. The default is 10 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum interval.

## Examples

This example configures the maximum time a client can wait before responding to a query to 5 seconds.

```
device(config)# ipv6 multicast max-response-time 5
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was modified to increase the range of the maximum response time from 1 through 10 seconds to 1 through 25 seconds. |

# ipv6 multicast mcache-age

Configures the time for an mcache to age out when it does not receive traffic.

## Syntax

**ipv6 multicast mcache-age** *num*

**no ipv6 multicast mcache-age** *num*

## Command Default

The mcache ages out after the default age-out interval, which is 180 seconds for ICX 7750, ICX 7450, and ICX 7250 devices.

## Parameters

*num*

Specifies the time, in seconds, the device should wait before stopping traffic to a port when a leave message is received The range is 60 through 3600 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default mcache age-out time.

You can set the time for a multicast cache (mcache) to age out when it does not receive traffic. Two seconds before an mcache is aged out, the device mirrors a packet of the mcache to the CPU to reset the age. If no data traffic arrives within two seconds, the mcache is deleted.

**NOTE**
Multicast mcache may not expire according to the configured time. You may notice a delay of 0 to 60 seconds over the configured value.

**NOTE**
On devices that support MAC-based MLD snooping (like the ICX 7750, ICX7450, and ICX 7250), more than one mcache can be mapped to the same destination MAC. When an mcache entry is deleted, the MAC entry may not be deleted. If you configure a lower value, the resource consumed by idle streams is quickly removed, but packets are mirrored to the CPU more frequently. Configure a higher value only when data streams are arriving consistently.

## Examples

This example configures the time for an mcache to age out to 180 seconds.

```
device(config)# ipv6 multicast mcache-age 180
```

# History

| Release version | Command history |
|---|---|
| 08.0.60 | Added note about multicast mcache expiry. |

# ipv6 multicast optimization

Enables or disables IP multicast (IPMC) hardware entry optimization for Layer 2 IPv6 multicast flows.

## Syntax

**ipv6 multicast optimization oif-listall**

**no ipv6 multicast optimization oif-list all**

## Command Default

Hardware entry optimization is disabled by default on ICX 7750 devices, and enabled by default on ICX 7450 and 7250 devices.

## Parameters

**oif-list**

Shares the Output Interface Lists across entries.

**all**

Specifies all types of Output Interface Lists.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command disables hardware entry optimization for Layer 2 IPv6 multicast flows. The command must be followed by the **write memory** command and the **reload** command for the changes to take effect.

## Examples

The following example enables hardware entry optimization for Layer 2 IPv6 multicast flows.

```
device(config)# ip multicast optimization oif-list all
device(config)# write memory
device(config)# exit
device# reload
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ipv6 multicast query-interval

Configures how often the device sends group membership queries when the multicast listening discovery (MLD) mode is set to active.

## Syntax

**ipv6 multicast query-interval** *interval*

**no ipv6 multicast query-interval** *interval*

## Command Default

Queries are sent every 125 seconds.

## Parameters

*interval*

Specifies the time, in seconds, between queries. The range is 10 through 3600 seconds. The default is 125 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the query interval to 125 seconds.

If the MLD mode is set to active, you can modify the query interval, which specifies how often the device sends group membership queries. When multiple queriers connect together, all queriers should be configured with the same interval.

## Examples

The following example configures the query interval to 120 seconds.

```
device#configure terminal
device(config)#ipv6 multicast query-interval 120
```

# ipv6 multicast report-control

Limits report forwarding within the same group to no more than once every 10 seconds.

## Syntax

**ipv6 multicast report-control**

**no ipv6 multicast report-control**

## Command Default

A device in passive mode forwards reports and leave messages from clients to the upstream router ports that are receiving queries.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default.

> **NOTE**
> This feature applies only to multicast listening discovery (MLD) version 1. The leave messages are not rate limited.

This rate-limiting does not apply to the first report answering a group-specific query.

Configure this command to alleviate report storms from many clients answering the upstream router query.

## Examples

This example limits the rate that reports are forwarded.

```
Device(config)#ipv6 multicast-report-control
```

# ipv6 multicast verbose-off

Turns off error or warning messages that are displayed when the device runs out of software resources or when it receives packets with the wrong checksum or groups.

## Syntax

**ipv6 multicast verbose-off**

**no ipv6 multicast verbose-off**

## Command Default

Messages are displayed.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default display of messages.

## Examples

This example turns off the display of messages.

```
device# configure terminal
device(config)# ipv6 multicast verbose-off
```

# ipv6 multicast version

Configures the multicast listening discovery (MLD) version for snooping globally.

## Syntax

**ipv6 multicast version** [ **1** | **2** ]

**no ipv6 multicast version**

## Command Default

MLD version 1 is configured.

## Parameters

**1**

Configures MLD version 1.

**2**

Configures MLD version 2.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the version to MLD version 1.

You can configure the MLD version for individual VLANs, or individual ports within VLANs. If no MLD version is specified for a VLAN, the globally configured MLD version is used. If an MLD version is specified for individual ports in a VLAN, those ports use that version instead of the version specified for the VLAN or the globally specified version. The default is MLD version 1.

## Examples

This example specifies MLD version 2 on a device.

```
Device(config)#ipv6 multicast version 2
```

# ipv6 multicast-boundary

Defines multicast boundaries for PIM-enabled interfaces.

## Syntax

**ipv6 multicast-boundary** *acl-spec*

**no ipv6 multicast-boundary** *acl-spec*

## Command Default

Boundaries are not defined.

## Parameters

*acl-spec*

Specifies the number or name identifying an access control list (ACL) that controls the range of group addresses affected by the boundary.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command removes the boundary on a PIM-enabled interface.

You can use standard ACL syntax to configure an access list.

## Examples

This example defines a boundary named MyAccessList for a PIM-enabled interface.

```
Device(config)# interface ethernet 1/2/2
Device(config-if-e1000-1/2)#ipv6 multicast-boundary MyAccessList
```

# ipv6 multicast-routing optimization

Enables or disables IP multicast (IPMC) entry optimization for Layer 3 IPv6 multicast flows.

## Syntax

**ipv6 multicast-routing optimization oif-list all**

**no ipv6 multicast-routing optimization oif-list all**

## Command Default

IPMC entry optimization is disabled by default on ICX 7750 devices, and enabled by default on ICX 7450 and 7250 devices.

## Parameters

**oif-list**

Shares the Output Interface Lists across entries.

**all**

Specifies all types of Output Interface Lists.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables IPMC entry optimization for IPv6 multicast flows. Multicast routing entries are deleted and recreated when optimization is enabled or disabled on all VRFs. The command must be followed by the **write memory** command and the **reload** command for the changes to take effect.

## Examples

The following example enables hardware entry optimization for IPv6 multicast flows.

```
device(config)# ipv6 multicast-routing optimization oif-list all
device(config)# write memory
device(config)# exit
device# reload
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ipv6 multicast-routing rpf-check mac-movement

Triggers Reverse Path Forwarding (RPF) check on MAC movement for directly connected sources and sends a MAC address movement notification to the Protocol Independent Multicast (PIM) module which results in PIM convergence.

## Syntax

**ipv6 multicast-routing rpf-check mac-movement**

**no ipv6 multicast-routing rpf-check mac-movement**

## Command Default

RPF check on MAC movement for directly connected sources is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

PIM convergence on MAC movement is applicable only in a topology where the multicast source port and PIM routers are in the same Layer 2 domain.

IPv6 PIM Dense mode is not supported for PIM convergence on MAC movement.

The **ipv6 multicast-routing rpf-check mac-movement** command is not supported on the Ruckus ICX 7250 devices.

The **no** form of the command disables RPF check on MAC movement for directly connected sources.

## Examples

The following example configures RPF check on MAC movement for directly connected sources.

```
device(config)# ipv6 multicast-routing rpf-check mac-movement
```

## History

| Release version | Command history |
|---|---|
| 08.0.10h | This command was introduced. |
| 08.0.30 | Support for the **ipv6 multicast-routing rpf-check mac-movement** command was added in 08.0.30 and later releases. |

# ipv6 nd dad attempts

Configures the number of consecutive neighbor solicitation messages that duplicate address detection (DAD) sends on an interface.

## Syntax

**ipv6 nd dad attempts** *number*

**no ipv6 nd dad attempts** *number*

## Command Default

By default, duplicate address detection sends three neighbor solicitation messages without any follow-up messages.

## Parameters

*number*

Specifies the number of consecutive neighbor solicitation messages that duplicate address detection sends on an interface. Valid values are 0 to 255. The default value is 3. Configuring a value of 0 disables duplicate address detection processing on the specified interface.

## Modes

Interface configuration mode

## Usage Guidelines

DAD is not currently supported with IPv6 tunnels. Make sure tunnel endpoints do not have duplicate IP addresses.

The **no** form of the command restores the number of messages to the default value of 3.

## Examples

The following example configures the number of consecutive neighbor solicitation messages that duplicate address detection sends on an interface to 100.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd dad attempts 2
```

# ipv6 nd managed-config-flag

Sets the managed address configuration flag.

## Syntax

**ipv6 nd managed-config-flag**

**no ipv6 nd managed-config-flag**

## Command Default

By default, the managed address configuration flag is not set in router advertisement messages.

## Modes

Interface configuration mode

## Usage Guidelines

An IPv6 router advertisement message includes the managed address configuration flag. This flag indicates to hosts on a local link if they should use the stateful autoconfiguration feature to get IPv6 addresses for their interfaces. If the flag is set, the hosts use stateful autoconfiguration to get addresses as well as non-IPv6-address information. If the flag is not set, the hosts do not use stateful autoconfiguration to get addresses and if the hosts can get non-IPv6-address information from stateful autoconfiguration is determined by the setting of the Other Stateful Configuration flag.

The **no** form of the command removes the managed address configuration flag from the router advertisement messages.

## Examples

The following example sets the managed address configuration flag.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd managed-config-flag
```

# ipv6 nd ns-interval

Configures the interval in milliseconds at which duplicate address detection sends a neighbor solicitation message on an interface.

## Syntax

**ipv6 nd ns-interval** *interval*

**no ipv6 nd ns-interval** *interval*

## Command Default

By default, duplicate address detection sends a neighbor solicitation message every 1000 milliseconds.

## Parameters

*interval*

Specifies the interval in milliseconds at which duplicate address detection sends a neighbor solicitation message on an interface. Valid values are 0 to 4294967295 milliseconds. The default value is 1000 milliseconds.

## Modes

Interface configuration mode

## Usage Guidelines

Ruckus does not recommend very short intervals in normal IPv6 operation. When a non-default value is configured, the configured time is both advertised and used by the router itself.

The **no** form of the command restores the interval to the default value of 1000 milliseconds.

## Examples

The following example configures the interval between the transmission of the two messages to 9 seconds.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd ns-interval 9000
```

# ipv6 nd other-config-flag

Configures the hosts can use stateful autoconfiguration to get non-IPv6-address information.

## Syntax

**ipv6 nd other-config-flag**

**no ipv6 nd other-config-flag**

## Command Default

By default, the other stateful configuration flags are not set in router advertisement messages.

## Modes

Interface configuration mode

## Usage Guidelines

The other stateful configuration flag indicates to hosts on a local link if they can get non-IPv6 address autoconfiguration information. If the flag is set, the hosts can use stateful autoconfiguration to get non-IPv6-address information.

When determining if hosts can use stateful autoconfiguration to get non-IPv6-address information, a set Managed Address Configuration flag overrides an unset Other Stateful Configuration flag. In this situation, the hosts can obtain nonaddress information. However, if the Managed Address Configuration flag is not set and the Other Stateful Configuration flag is set, then the setting of the Other Stateful Configuration flag is used.

The **no** form of the command other stateful configuration flag.

## Examples

The following example sets the other stateful configuration flag.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ipv6 nd other-config-flag
```

# ipv6 nd prefix-advertisement

Configures the prefixes to be included in router advertisement messages.

## Syntax

**ipv6 nd prefix-advertisement** *ipv6-address valid-lifetime preffered-lifetime* [ **auto-config** ] [ **onlink** ]

**no ipv6 nd prefix-advertisement** *ipv6-address valid-lifetime preffered-lifetime* [ **auto-config** ] [ **onlink** ]

## Command Default

By default, router advertisement messages include prefixes configured as addresses on router interfaces using the **ipv6 address** command.

## Parameters

*ipv6-address*

Specifies the IPv6 addres in hexadecimal using 16-bit values between colons as documented in RFC 2373 along with the prefix length in the format X:X::X:X/M.

*valid-lifetime*

Configures the time interval (in seconds) in which the specified prefix is advertised as valid. Valid values are 0 through 4294967295 seconds. The default is 2592000 seconds (30 days). When the timer expires, the prefix is no longer considered to be valid.

*preffered-lifetime*

Configures the time interval (in seconds) in which the specified prefix isadvertised as preferred. Valid values are 0 through 4294967295 seconds. The default is 604800 seconds (7 days). When the timer expires, the prefix is no longer considered to be preferred.

**auto-config**

If this flag is set, the stateless auto configuration feature can use the specified prefix in the automatic configuration of 128-bit IPv6 addresses for hosts on the local link, provided the specified prefix is aggregatable, as specified in RFC 2374.

**onlink**

If this flag is set, the specified prefix is assigned to the link upon which it is advertised. Nodes sending traffic to addresses that contain the specified prefix consider the destination to be reachable on the local link.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command removes a prefix from the router advertisement messages sent from a particular interface.

# Examples

The following example configures to advertise the prefix 2001:DB8:a487:7365::/64 in router advertisement messages sent out on Ethernet interface 1/3/1 with a valid lifetime of 1000 seconds, a preferred lifetime of 800 seconds, and the Onlink and Autoconfig flags set.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd prefix-advertisement 2001:DB8:a487:7365::/64 1000 800 onlink
autoconfig
```

# ipv6 nd ra-dns-server

Configures the IPv6 router advertisement (RA) of Domain Name System (DNS) server addresses and the lifetime multiplier on an interface.

## Syntax

**ipv6 nd ra-dns-server** *ipv6-address* [ **lifetime-multiplier** *decimal* ]

**no ipv6 nd ra-dns-server** *ipv6-address* [ **lifetime-multiplier** *decimal* ]

## Command Default

Recursive DNS server (RDNSS) address and lifetime multiplier information is not configured. The DNS server is not advertised in IPv6 RA messages.

## Parameters

*ipv6-address*

Specifies the IPv6 address of the DNS server to be advertised in RA messages.

**lifetime-multiplier** *decimal*

Specifies the value of the maximum RA interval (the maximum time allowed between sending unsolicited RA messages for DNS name resolution). The calculated lifetime value = configured value * max RA interval. Valid values range from 1 through 3. The default is 3 (that is, three times the configured maximum RA interval).

## Modes

Interface configuration mode

## Usage Guidelines

You can configure a maximum of 15 RDNSS addresses and corresponding lifetime multiplier values in a given instance.

The **no** form of the command removes the configured DNS server address and lifetime multiplier value.

## Examples

The following example configures a DNS server with the IPv6 address 2001::1 to be advertised in RA messages, with a lifetime multiplier of 2.

```
device# configure terminal
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# ipv6 nd ra-dns-server 2001::1 lifetime-multiplier 2
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# ipv6 nd ra-domain-name

Configures the IPv6 router advertisement (RA) of Domain Name System (DNS) suffixes and the lifetime multiplier on an interface.

## Syntax

**ipv6 nd ra-domain-name** *string* [ **lifetime-multiplier** *decimal* ]

**no ipv6 nd ra-domain-name** *string* [ **lifetime-multiplier** *decimal* ]

## Command Default

The DNS suffix is not advertised in IPv6 RA messages.

## Parameters

*string*

Specifies the domain name of the DNS suffix.

**lifetime-multiplier** *decimal*

Specifies the value of the maximum RA interval (the maximum time that can be allowed between sending unsolicited RA messages for DNS name resolution). The calculated lifetime value = configured value * max RA interval. Valid values range from 1 through 3. The default is 3 (that is. three times the configured maximum RA interval).

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables the advertisement of the DNS suffix in IPv6 RA messages.

## Examples

The following example configures the domain name of a DNS suffix as “abc.net” and sets a lifetime multiplier value of 1 for an Ethernet interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# ipv6 nd ra-domain-name abc.net lifetime-multiplier 1
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# ipv6 nd ra-hop-limit

Sets the hop limit for router advertisement messages.

## Syntax

**ipv6 nd ra-hop-limit** *number*

**no ipv6 nd ra-hop-limit** *number*

## Command Default

The default hop is 64.

## Parameters

*number*

Specifies the number of hops. Valid values are 0 to 255. The default value is 64.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the commands resets the number of hops to the default value of 64.

## Examples

The following example sets the number of hops to 100.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd ra-hop-limit 100
```

# ipv6 nd ra-interval

Configures the interval at which an interface sends router advertisement messages.

## Syntax

**ipv6 nd ra-interval** [ **range** *min-interval max-interval* | *interval* ]

**no ipv6 nd ra-interval** [ **range** *min-interval max-interval* | *interval* ]

## Command Default

By default, an interface sends a router advertisement message every 200 seconds.

## Parameters

**range** *min-interval max-interval*

Configures an interval range. The min-range-value specifies the minimum number of seconds allowed between sending unsolicited multicast router advertisements from the interface. The default is 0.33 times the max-range-value if the max-range-value is greater than or equal to 9 seconds. Otherwise, the default is the value specified by the max-range-value. The min-range-value can be a number between -3 - (.75 x max range value ). The max-range-value parameter specifies the maximum number of seconds allowed between sending unsolicited multicast router advertisements from the interface. This number can be between 4 - 1800 seconds and must be greater than the min-range-value x 1.33. The default is 600 seconds.

*interval*

Configures the interval. Valid values are 3 to1800 seconds. The default is 200 seconds. The actual RA interval will be from .5 to 1.5 times the configured or default value.

## Modes

Interface configuration mode

## Usage Guidelines

Ruckus recommends that the interval between router advertisement transmission be less than or equal to the router lifetime value if the router is advertised as a default router.

The **no** form of the command resets the interface at which an interface sends a router advertisement message to 200 seconds.

## Examples

The following example configures the interval at which an interface sends a router advertisement message as 300 seconds.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd ra-interval 300
```

The following example configures the interval at which an interface sends a router advertisement message to a range.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd ra-interval range 33 55
```

# ipv6 nd ra-lifetime

Configures the value (in seconds) indicates if the router is advertised as a default router on an interface.

## Syntax

**ipv6 nd ra-lifetime** *time*

**no ipv6 nd ra-lifetime** *time*

## Command Default

By default, the router lifetime value included in router advertisement messages sent from an interface is 1800 seconds.

## Parameters

*time*

Specifies the value (in seconds) indicates if the router is advertised as a default router on an interface. Valid values are 0 to 9000 seconds. The default is 1800 seconds. If you set the value of this parameter to 0, the router is not advertised as a default router on an interface.

## Modes

Interface configuration mode

## Usage Guidelines

The "router lifetime" value, which is included in router advertisements sent from a particular interface.

If you set this parameter to a value that is not 0, the router is advertised as a default router on the interface.

Ruckus recommends that the interval between router advertisement transmission be less than or equal to the router lifetime value if the router is advertised as a default router.

The **no** form of the command resets the value to 1800 seconds.

## Examples

The following example configures the router lifetime value to 1900 seconds on Ethernet interface 1/3/1.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd ra-lifetime 1900
```

# ipv6 nd reachable-time

Configures the duration that a router considers a remote IPv6 node reachable.

## Syntax

**ipv6 nd reachable-time** *duration*

**no ipv6 nd reachable-time** *duration*

## Command Default

By default, a router interface uses the value of 30 seconds.

## Parameters

*duration*

Specifies the duration (in seconds) that a router considers a remote IPv6 node reachable. Valid values are 0 to 3600 seconds. The default is 30 seconds.

## Modes

Interface configuration mode

## Usage Guidelines

The router advertisement messages sent by a router interface include the amount of time specified by the ipv6 nd reachable-time command so that nodes on a link use the same reachable time duration. By default, the messages include a default value of 0.

Ruckus does not recommend configuring a short reachable time duration, because a short duration causes the IPv6 network devices to process the information at a greater frequency.

The actual reachable time will be from 0.5 to 1.5 times the configured or default value.

The **no** form of the command resets the duration that a router considers a remote IPv6 node reachable as 30 seconds.

## Examples

The following example configures the reachable time of 40 seconds for Ethernet interface 1/3/1.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd reachable-time 40
```

# ipv6 nd router-preference

Configures the IPv6 router advertisement preference value to low or high (medium is the default). IPv6 router advertisement preference enables IPv6 router advertisement (RA) messages to communicate default router preferences from IPv6 routers to IPv6 hosts in network topologies where the host has multiple routers on its Default Router List.

## Syntax

**ipv6 nd router-preference** [ **low** | **medium** | **high** ]

**no ipv6 nd router-preference** [ **low** | **medium** | **high** ]

## Command Default

The IPv6 router advertisement preference value is set to medium.

## Parameters

**low**

The two-bit signed integer (11) indicating the preference value "low".

**medium**

The two-bit signed integer (00) indicating the preference value "medium". This is the default preference value.

**high**

The two-bit signed integer (01) indicating the preference value "high".

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form disables IPv6 router preference.

## Examples

The following example configures IPv6 RA preference for IPv6 routers:

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e10000-1/2/3)# ipv6 nd router-preference low
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# ipv6 nd suppress-ra

Disables the sending of router advertisement messages on an interface.

## Syntax

**ipv6 nd suppress-ra**

**no ipv6 nd suppress-ra**

## Command Default

Sending of router advertisement messages is enabled on Ethernet interfaces and disabled on non-LAN interfaces.

## Modes

Interface configuration mode

## Usage Guidelines

If IPv6 unicast routing is enabled on an Ethernet interface, by default, this interface sends IPv6 router advertisement messages. However, by default, non-LAN interface types, for example, tunnel interfaces, do not send router advertisement messages.

The **no** form of the command enables the sending of router advertisement messages on a interface.

## Examples

The following example disables the sending of router advertisement messages on an Ethernet interface.

```
device(config)# interface ethernet 1/3/1
device(config-if-e1000-1/3/1)# ipv6 nd suppress-ra
```

The following example enables the sending of router advertisement messages on a tunnel interface.

```
device(config)# interface tunnel 1
device(config-tnif-1)# no ipv6 nd suppress-ra
```

# ipv6 nd suppress-ra address

Suppresses the advertisement of specified IPv6 addresses for router advertisement (RA) messages on an interface.

## Syntax

**ipv6 nd suppress-ra address** { **all** | *ipv6-address* }

**no ipv6 nd suppress-ra address** { **all** | *ipv6-address* }

## Command Default

IPv6 addresses are not suppressed.

## Parameters

**all**
Specifies all IPv6 addresses.

*ipv6-address*
Specifies an IPv6 address.

## Modes

Interface configuration mode

## Usage Guidelines

Prefix information in RA messages includes the IPv6 addresses configured on the interface.

The **no** form of the command restores the default so that IPv6 addresses are not suppressed in RA messages.

## Examples

The following example suppresses all IPv6 addresses configured on the interface in RA messages.

```
device# configure terminal
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# ipv6 nd suppress-ra address all
```

The following example suppresses the IPv6 address 2001::1 in RA messages. All other IPv6 addresses configured on the interface are advertised.

```
device# configure terminal
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# ipv6 nd suppress-ra address 2001::1
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# ipv6 neighbor

Adds a static entry to the IPv6 neighbor discovery cache.

## Syntax

**ipv6 neighbor** *ipv6-address* [ **ve** *ve-num* ] **ethernet** *stack**/** slot**/**port link-layer-address*

**no ipv6 neighbor** *ipv6-address* [ **ve** *ve-num* ] **ethernet** *stack**/** slot**/**port link-layer-address*

## Command Default

A static entry is not added to the IPv6 neighbor discovery cache.

## Parameters

*ipv6-address*
: Specifies the IPv6 address of the neighbor.

**ve** *ve-num*
: Specifies the outgoing interface type as VE.

**ethernet** *stack**/** slot**/**port*
: Specifies the outgoing interface type as Ethernet. If you specify VE, specify the Ethernet interface associated with the VE.

*link-layer-address*
: Specifies the 48-bit hardware address of the neighbor.

## Modes

Global configuration mode

## Usage Guidelines

In some special cases, a neighbor cannot be reached using the neighbor discovery feature. In this situation, you can add a static entry to the IPv6 neighbor discovery cache, which causes a neighbor to be reachable at all times without using neighbor discovery. (A static entry in the IPv6 neighbor discovery cache functions like a static ARP entry in IPv4.)

A port that has a statically assigned IPv6 entry cannot be added to a VLAN.

Static neighbor configurations will be cleared on secondary ports when a LAG is formed.

If you attempt to add an entry that already exists in the neighbor discovery cache, the software changes the already existing entry to a static entry.

The **no** form of the command removes a static IPv6 entry from the IPv6 neighbor discovery cache.

# Examples

The following example adds a static entry for a neighbor with the IPv6 address 2001:DB8:2678:47b and linklayer address 0000.002b.8641 that is reachable through Ethernet interface 1/3/1.

```
device(config)# ipv6 neighbor 2001:DB8:2678:47b ethernet 1/3/1 0000.002b.8641
```

# ipv6 neighbor inspection

Configures the static neighbor discovery (ND) inspection entries.

## Syntax

**ipv6 neighbor inspection** *ipv6-address mac-address*

**no ipv6 neighbor inspection** *ipv6-address mac-address*

## Command Default

Static ND inspection entries are not configured.

## Parameters

*ipv6-address*
Configures the IPv6 address of the host.

*mac-address*
Configures the MAC address of the host.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

Use the **ipv6 neighbor inspection** command to manually configure static ND inspection entries for hosts on untrusted ports. During ND inspection, the IPv6 address and MAC address entries in the ND inspection table are used to validate the packets received on untrusted ports.

The **no** form of the command disables static ND inspection entries.

## Examples

The following example displays the configuration of a static ND inspection entry.

```
device(config)# ipv6 neighbor inspection 2001::1 0000.1234.5678
```

The following example displays the configuration of a static ND inspection entry for VRF 3.

```
device(config)# vrf 3
device(config-vrf-3)# ipv6 neighbor inspection 2001::100 0000.0000.4567
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# ipv6 neighbor inspection vlan

Configures and enables Neighbor Discovery (ND) inspection on a VLAN, or a range of VLANs, to inspect the IPv6 packets from untrusted ports.

## Syntax

**ipv6 neighbor inspection vlan** *vlan-id* [ **to** *vlan-id* ... ]

**no ipv6 neighbor inspection vlan** *vlan-id* [ **to** *vlan-id* ... ]

## Command Default

IPv6 ND inspection is not enabled.

## Parameters

*vlan-id*

Configures the ID of the VLAN.

**to** *vlan-id*

Specifies a range of VLANs.

## Modes

Global configuration mode

VRF configuration mode

## Usage Guidelines

When you configure this command, IPv6 packets from untrusted ports on the VLAN undergo ND inspection.

All VLANs included in the range when using the **to** keyword must be valid VLANs. Otherwise an error will occur.

The **no** form of the command disables ND inspection.

## Examples

The following example enables ND inspection on VLAN 10.

```
device# configure terminal
device(config)# ipv6 neighbor inspection vlan 10
```

The following example enables ND inspection on VLAN 10 of VRF 3.

```
device# configure terminal
device(config)# vrf 3
device(config-vrf-3)# ipv6 neighbor inspection vlan 10
```

The following example configures VLANs 100 through 150, VLAN 160, and VLANs 170 through 200 and enables ND inspection on all of the configured VLANs.

```
device# configure terminal
device(config)# vlan 100 to 150
device(config-mvlan-100-150)# exit
device(config)# vlan 150 to 200
device(config-mvlan-150-200)# exit
device(config)# ipv6 neighbor inspection vlan 100 to 150 160 170 to 200
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.80 | The **to** keyword was added to enable ND inspection on a range of VLANs by using a single command. |

# ipv6 ospf active

Sets a specific OSPFv3 interface to active.

## Syntax

**ipv6 ospf active**

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use the **ipv6 ospf active** command on each interface participating in adjacency formation. This command overrides the global passive setting on that interface, and enables transmission of OSPFv3 control packets.

## Examples

The following example sets a specific OSPFv3 virtual Ethernet (VE) interface to active.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf active
```

# ipv6 ospf area

Enables OSPFv3 on an interface.

## Syntax

**ipv6 ospf area** *area-id* | *ip-addr*

**no ipv6 ospf area**

## Command Default

OSPFv3 is disabled.

## Parameters

*area-id*

Area ID in dotted decimal or decimal format.

*ip-addr*

Area ID in IP address format.

## Modes

Interface subtype configuration mode

## Usage Guidelines

This command enables an OSPFv3 area on the interface to which you are connected.

The **no** form of the command disables OSPFv3 on this interface.

## Examples

The following example enables a configured OSPFv3 area named 0 on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf area 0
```

# ipv6 ospf authentication

Configures HMAC-SHA-1 or HMAC-SHA-256 authentication for Open Shortest Path First version 3 (OSPFv3).

## Syntax

**ipv6 ospf authentication** { **hmac-sha-1** | **hamac-sha-256**} **key-id** *key-id-val* **key** *key-string*

**no ipv6 ospf authentication** [ **HMAC-SHA-1** | **HMAC-SHA-256**] **key-id** *key-id-val* **key** *key-string*

## Command Default

HMAC-SHA-1 or HMAC-SHA-256 authentication is disabled by default.

## Parameters

**hmac-sha-1**
Specifies the HMAC-SHA-1 authentication.

**hmac-sha-256**
Specifies the HMAC-SHA-256 authentication.

**key-id** *key-id-val*
Identifies the number of the HMAC-SHA-1 or HMAC-SHA-256 algorithm. The number can be from 1 through 255.

**key** *key-string*
Sets the corresponding key string to be used with the HMAC-SHA-1 or HMAC-SHA-256 algorithm.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the HMAC-SHA-1 or HMAC-SHA-256 authentication configuration on the OSPFv3 interface to which you are connected.

The **no** form of the command removes the HMAC-SHA-1 or HMAC-SHA-256 authentication configuration from the OSPFv3 interface.

## Examples

The following example sets HMAC-SHA-1 authentication with key ID 10 and the password key "mypasswordkey", on the OSPFv3 interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf authentication hmac-sha-1 key-id 10 key mypasswordkey
```

# History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ipv6 ospf authentication disable

Removes the authentication configuration settings on a specific interface in an Open Shortest Path First version 3 (OSPFv3) area.

## Syntax

**ipv6 ospf authentication disable**

## Modes

Interface subtype configuration mode

## Usage Guidelines

Where an area is configured with area authentication, all interfaces within the area are configured to use these authentication parameters. This command removes the authentication configuration settings on a specific interface within the area.

## Examples

The following example removes the authentication configuration settings on the selected interface within the OSPFv3 area.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-11)# ipv6 ospf authentication disable
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ipv6 ospf authentication ipsec

Specifies IP security (IPsec) as the authentication type for an OSPFv3 interface.

## Syntax

**ipv6 ospf authentication ipsec key-add-remove-interval** *interval*

**no ipv6 ospf authentication ipsec key-add-remove-interval** *interval*

## Command Default

Disabled.

## Parameters

**key-add-remove-interval** *interval*

Specifies the OSPFv3 authentication key add-remove interval. Valid values range from decimal numbers 0 through 14400. The default is 300.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command removes IPsec authentication from the interface.

## Examples

The following example enables IPsec on a specified OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf area 0
device(config-vif-1)# ipv6 ospf authentication ipsec
```

The following example sets the OSPFv3 authentication key add-remove interval to 480.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf area 0
device(config-vif-1)# ipv6 ospf authentication ipsec key-add-remove-interval 480
```

# ipv6 ospf authentication ipsec disable

Disables IP security (IPsec) services on an OSPFv3 interface.

## Syntax

**ipv6 ospf authentication ipsec disable**

**no ipv6 ospf authentication ipsec disable**

## Command Default

Authentication is disabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to disable IPsec if it is enabled on the interface. Packets that are sent out will not be IPSec encapsulated and the received packets which are IPSec encapsulated will be dropped.

The **no** form of the command re-enables IPSec on the interface if IPsec is already configured on the interface.

## Examples

The following example disables IPsec on a specific OSPFv3 interface where IPsec is already enabled.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf authentication ipsec disable
```

# ipv6 ospf authentication ipsec spi

Specifies the IP security (IPsec) security policy index (SPI) value for an OSPFv3 interface.

## Syntax

**ipv6 ospf authentication ipsec spi** *value* **esp sha1** *key* [ **no-encrypt** ] *key* }

**no ipv6 ospf authentication spi**

## Command Default

Authentication is disabled.

The 40-hexadecimal character key is encrypted by default. Use the **no-encrypt** parameter to disable encryption.

## Parameters

**ipsec**

Specifies IPsec as the authentication protocol.

**spi**

Specifies the Security Policy Index (SPI).

*value*

Specifies the SPI value. Valid values range from decimal numbers 256 through 4294967295. The near-end and far-end values must be the same.

**esp**

Specifies Encapsulating Security Payload (ESP) as the protocol to provide packet-level security. This is the only option currently available.

**sha1**

Enables Hashed Message Authentication Code (HMAC) Secure Hash Algorithm 1 (SHA-1) authentication.

*key*

Number used in the calculation of the message digest. The 40 hexadecimal character key is stored in encrypted format by default.

**no-encrypt**

The 40-character key is not encrypted upon either its entry or its display.

*key*

The 40 hexadecimal character key.

## Modes

Interface subtype configuration mode

# Usage Guidelines

The 40 hexadecimal character key is encrypted by default. The system adds the following in the configuration to indicate that the key is encrypted:

- encrypt = the key string uses proprietary simple cryptographic 2-way algorithm
- encryptb64 = the key string uses proprietary base64 cryptographic 2-way algorithm

To change an existing key, you must specify a different SPI value to that of the value already configured.

The **no** form of the command removes the SPI value from the interface.

# Examples

The following example enables ESP and HMAC-SHA-1 on a specified OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf area 0
device(config-vif-1)# ipv6 ospf authentication ipsec spi 512 esp sha1
abcef12345678901234fedcba098765432109876
```

# ipv6 ospf authentication key-activation-wait-time

Configures the time before an authentication key change is activated for an Open Shortest Path First version 3 (OSPFv3) interface.

## Syntax

**ipv6 ospf authentication key-activation-wait-time** *wait-time*

**no ipv6 ospf authentication key-activation-wait-time** *wait-time*

## Parameters

*wait-time*

Specifies the time before an authentication key change takes place. The wait time can be set from 0 through 14400 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the wait time before an authentication key change takes place on the interface to which you are connected.

The **no** form of the command resets the wait time to the default of 300 seconds.

## Examples

The following example sets the wait time before an authentication key change to 600 seconds on the OSPFv3 interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf authentication key-activation-wait-time 600
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ipv6 ospf authentication keychain

Configures Open Shortest Path First version 3 (OSPFv3) authentication using the keychain authentication module.

## Syntax

**ipv6 ospf authentication keychain** *keychain-name*

**no ipv6 ospf authentication keychain** *keychain-name*

## Parameters

*keychain-name*

Specifies the name of the keychain that OSPFv3 uses to authenticate the packets.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The keychain authentication module provides OSPFv3 protocol the option to automatically change the key ID and cryptographic algorithm without manual intervention.

With this configuration, OSPFv3 requests the keychain authentication module for all active keys in the keychain and selects the keys for sending and accepting the packets.

The **no** form of the command removes keychain authentication from the OSPFv3 interface configuration.

## Examples

The following example configures OSPFv3 to use the keychain authentication module with the "xtreme" keychain.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-11)# ipv6 ospf authentication keychain xtreme
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ipv6 ospf authentication rfc6506

Configures authentication in accordance with RFC 6506 for Open Shortest Path First version 3 (OSPFv3).

## Syntax

**ipv6 ospf authentication rfc6506**

**no ipv6 ospf authentication rfc6506**

## Command Default

RFC 6506 authentication is disabled by default.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset authentication in accordance with RFC 6506 on the OSPFv3 interface to which you are connected. This may be required for backward compatibility. Although RFC 6506 is superseded by RFC 7166, some vendors continue to support RFC 6506. To ensure interoperability with vendor equipment that supports RFC 6506, use this command in conjunction with the required authentication options, as shown in the example below.

The **no** form of the command removes the RFC 6506 authentication configuration from the OSPFv3

## Examples

The following example sets HMAC-SHA-1 authentication, in accordance with RFC 6506, on the OSPFv3 interface. HMAC-SHA-1 authentication is enabled using key-id "1", key "0 1234567890123456789", and a key activation wait time of 5 seconds.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf authentication rfc6506
device(config-vif-1)# ipv6 ospf authentication hmac-sha-1 key-id 1 key 0 1234567890123456789
device(config-vif-1)# ipv6 ospf authentication key-activation-wait-time 5
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# ipv6 ospf cost

Configures cost for a specific OSPFv3 interface.

## Syntax

**ipv6 ospf cost** *value*

**no ipv6 ospf cost**

## Command Default

Cost value is 1.

## Parameters

*value*

Cost value. Valid values range from 1 through 65535. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Use this command to set or reset the OSPFv3 cost on the interface. If the cost is not configured with this command, OSPFv3 calculates the value from the reference and interface bandwidths.

For more information, refer to the **auto-cost reference-bandwidth** command.

The **no** form of the command disables the configured cost.

## Examples

The following example sets the cost to 620 on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-11)# ipv6 ospf cost 620
```

# ipv6 ospf dead-interval

Specifies the time period for which a neighbor router waits for a hello packet from the device before declaring the router down.

## Syntax

**ipv6 ospf dead-interval** *interval*

**no ipv6 ospf dead-interval**

## Command Default

The specified time period is 40 seconds.

## Parameters

*interval*

Dead interval in seconds. Valid values range from 2 through 65535 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the dead interval, the hello interval is automatically changed to a value that is one fourth that of the new dead interval, unless the hello interval is also explicitly configured using the **ipv6 ospf hello-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is ¼ times that of the dead interval.
- If the OSPF hello interval and dead interval are set to more aggressive levels than 1:4 seconds respectively, the OSPF protocol might flap when the **write memory** command is used or in the case of any high CPU.

The **running-config** command displays only explicitly configured values of the hello interval, which means that a value that was automatically changed as the result of a dead-interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the dead interval to 80 on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf dead-interval 80
```

# ipv6 ospf hello-interval

Sets the length of time between the transmission of hello packets that an interface sends to neighbor routers.

## Syntax

**ipv6 ospf hello-interval** *interval*

**no ipv6 ospf hello-interval**

## Command Default

The length of time between the transmission of hello packets is set to 10 seconds.

## Parameters

*interval*

Hello interval in seconds. Valid values range from 1 through 65535 seconds. The default is 10.

## Modes

Interface subtype configuration mode

## Usage Guidelines

If you change the hello interval, the dead interval is automatically changed to a value that is four times that of the new hello interval, unless the dead interval is also explicitly configured using the **ipv6 ospf dead-interval** command.

The recommended setting is that:

- The dead interval is four times that of the hello interval.
- The hello interval is ¼ times that of the dead interval.
- If the OSPF hello interval and dead interval are set to more aggressive levels than 1:4 seconds respectively, the OSPF protocol might flap when the **write memory** command is used or in the case of any high CPU.

The **running-config** command displays only explicitly configured values of the dead interval, which means that a value that was automatically changed as the result of a hello interval change is not displayed.

The **no** form of the command restores the default value.

## Examples

The following example sets the hello interval to 20 on a specific OSPFv3 virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface ve 1
device(config-vif-1)# ipv6 ospf hello-interval 20
```

# ipv6 ospf hello-jitter

Sets the allowed jitter between HELLO packets.

## Syntax

**ipv6 ospf hello-jitter** *interval*

**no ipv6 ospf hello-jitter**

## Parameters

*jitter*

Allowed interval between hello packets.Valid values range from 1 through 50 percent (%).

## Modes

Interface subtype configuration mode

## Usage Guidelines

The hello interval can vary from the configured hello-interval to a maximum of percentage value of configured jitter.

## Examples

The following example sets the hello jitter to 20 on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf hello-jitter 20
```

# ipv6 ospf instance

Specifies the number of OSPFv3 instances running on an interface.

## Syntax

**ipv6 ospf instance** *instanceID*

**no ipv6 ospf instance**

## Parameters

*instanceID*

Instance identification number. Valid values range from 0 through 255.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

## Examples

The following example sets the number of IPv6 OSPF instances to 35 on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf instance 35
```

# ipv6 ospf mtu-ignore

Enables or disables maximum transmission unit (MTU) match checking.

## Syntax

**ipv6 ospf mtu-ignore**

**no ipv6 ospf mtu-ignore**

## Command Default

Enabled.

## Modes

Interface subtype configuration mode

## Usage Guidelines

In default operation, the IP MTU on both sides of an OSPFv3 link must be the same, and a check of the MTU is performed when Hello packets are first exchanged.

The **no** no form of the command disables MTU-match checking on a specific interface.

## Examples

The following example disables MTU-match checking on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# no ipv6 ospf mtu-ignore
```

The following example enables MTU-match checking on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf mtu-ignore
```

# ipv6 ospf network

Configures network type.

## Syntax

**ipv6 ospf network** { **broadcast** | **point-to-point** }

**no ipv6 ospf network**

## Command Default

Network type is broadcast for Ethernet and VE interfaces. Network type is point-to-point for tunnel and GRE interfaces.

## Parameters

**broadcast**
Network type is broadcast, such as Ethernet.

**point-to-point**
Network type is point-to-point.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Point-to-point can support unnumbered links, which requires less processing by OSPFv3.

The **no** form of the command removes the network-type configuration.

**NOTE**
The network type non-broadcast is not supported at this time.

## Examples

The following example configures an OSPFv3 point-to-point link on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf network point-to-point
```

The following example configures an OSPFv3 broadcast link on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf network broadcast
```

# ipv6 ospf passive

Sets a specific OSPFv3 interface to passive.

## Syntax

**ipv6 ospf passive**

**no ipv6 ospf passive**

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **ipv6 ospf passive** command disables transmission of OSPFv3 control packets on that interface. OSPFv3 control packets received on a passive interface are discarded.

The **no** form of the command sets an interface back to active.

## Examples

The following example sets a specific OSPFv3 virtual Ethernet (VE) interface to passive.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf passive
```

# ipv6 ospf priority

Configures priority for designated router (DR) election and backup designated routers (BDRs) on the interface you are connected to.

## Syntax

**ipv6 ospf priority** *value*

**no ipv6 ospf priority**

## Command Default

The value is set to 1.

## Parameters

*value*

Priority value. Valid values range from 0 through 255. The default is 1.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The OSPFv3 router assigned the highest priority becomes the designated router, and the OSPFv3 router with the second-highest priority becomes the backup router.

The **no** form of the command restores the default value.

## Examples

The following example sets a priority of 4 for the OSPFv3 router that is connected to an OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf priority 4
```

# ipv6 ospf retransmit-interval

Configures the retransmit interval. The retransmit interval is the time between Link-State Advertisement (LSA) retransmissions to adjacent routers for a given interface.

## Syntax

**ipv6 ospf retransmit-interval** *interval*

**no ipv6 ospf retransmit-interval**

## Command Default

The interval is 5 seconds.

## Parameters

*interval*

Retransmit interval in seconds. Valid values range from 0 through 3600 seconds. The default is 5.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command resets the retransmit interval to its default.

## Examples

The following example sets the retransmit interval to 8 for all OSPFv3 devices on an OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf retransmit-interval 8
```

# ipv6 ospf suppress-linklsa

Suppresses link LSA advertisements.

## Syntax

**ipv6 ospf suppress-linklsa**

**no ipv6 ospf suppress-linklsa**

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the defaults where link LSA advertisements are not suppressed.

## Examples

The following example suppresses link LSAs from being advertised on devices on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf suppress-linklsa
```

# ipv6 ospf transmit-delay

Configures transmit delay for link-update packets. The transmit delay is the estimated time required for OSPFv3 to send link-state update packets on the interface to which you are connected.

## Syntax

**ipv6 ospf transmit-delay** *value*

**no ipv6 ospf transmit-delay**

## Command Default

The transmit delay is set to 1 second.

## Parameters

*value*

Transmit delay in seconds. Valid values range from 0 through 3600 seconds.

## Modes

Interface subtype configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

## Examples

The following example sets a transmit delay of 25 seconds for devices on a specific OSPFv3 Virtual Ethernet (VE) interface.

```
device# configure terminal
device(config)# interface interface ve 1
device(config-vif-1)# ipv6 ospf transmit-delay 25
```

# ipv6 pim border

Configures an interface to be on a PIM Sparse domain border.

## Syntax

**ipv6 pim border**

**no ipv6 pim border**

## Command Default

The interface is not configured as a border device.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command removes the boundary on a PIM-enabled interface.

You must enable PIM globally before you enable it on an interface.

## Examples

This example configures Ethernet interface 3/2/4 to be on a PIM Sparse domain border.

```
device(config) interface ethernet 3/2/4
Device(config-if-e10000-3/2/4)# ipv6 pim border
```

# ipv6 pim dr-priority

Configures the designated router (DR) priority on IPv6 interfaces.

## Syntax

**ipv6 pim dr-priority** *priority-value*

**no ipv6 pim** *priority-value*

## Command Default

The DR priority value is 1.

## Parameters

*priority-value*

Specifies the DR priority value as an integer. The range is 0 through 65535. The default is 1.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command restores the default DR priority value, 1.

You must enable PIM globally before you enable it on an interface.

If more than one device has the same DR priority on a subnet (as in the case of default DR priority on all), the device with the numerically highest IPv6 address on that subnet is elected as the DR.

The DR priority information is used in the DR election only if all the PIM devices connected to the subnet support the DR priority option. If at least one PIM device on the subnet does not support this option, the DR election falls back to the backwards compatibility mode in which the device with the numerically highest IPv6 address on the subnet is declared the DR regardless of the DR priority values.

## Examples

This example configures a DR priority value of 50 on Ethernet interface 3/2/4.

```
device(config) interface ethernet 3/2/4
Device(config-if-e10000-3/2/4)# ipv6 pim dr-priority 50
```

This example configures a DR priority value of 50 on a virtual Ethernet interface.

```
Device(config)# interface ve 10
Device(config-vif-10)# ipv6 pim dr-priority 50
```

# ipv6 pim neighbor-filter

Determines which devices can become PIM neighbors.

## Syntax

**ipv6 pim neighbor-filter** *acl-name*

**no ipv6 pim** *acl-name*

## Command Default

Neighbor filtering is not applied on the interface.

## Parameters

*acl-name*

Specifies the access-control list (ACL)that identifies the devices you want to permit and deny participation in PIM.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command removes any neighbor filtering applied on the interface.

You must enable PIM globally before you enable it on an interface.

You can configure the **ipv6 pim neighbor-filter** command in either Dense mode (DM) or Sparse mode (SM).

Configure the **access-list** command to create an ACL defining the devices you want to permit and deny participation in PIM.

## Examples

This example prevents the host from becoming a PIM neighbor on interface Ethernet 1/3/24.

```
Device(config)# interface ethernet 1/3/24
Device(config-if-e10000-1/3/24)# ipv6 pim neighbor-filter
```

This example configures an ACL named 10 to deny a host and then prevents that host, 1001::1/96, identified in that ACL from becoming a PIM neighbor on interface Ethernet 1/3/24.

```
Device(config)# access-list 10  deny host 1001::1/96
Device(config)# access-list 10 permit any
Device(config)# interface ethernet 1/3/24
Device(config-if-e10000-1/3/24)# ipv6 pim neighbor-filter 10
```

# History

| Release version | Command history |
|---|---|
| 8.0.20a | This command was introduced. |

# ipv6 pim-sparse

Enables PIM Sparse on an IPv6 interface.

## Syntax

**ipv6 pim-sparse**

**no ipv6 pim-sparse**

## Command Default

PIM Sparse is not enabled on the IPv6 interface.

## Modes

Interface configuration mode

## Usage Guidelines

The **no ipv6 pim-sparse** command removes the PIM sparse configuration from the IPv6 interface.

## Examples

This example adds an IPv6 interface to port 1/2/2, then enables PIM Sparse on the interface.

```
Device(config)# interface ethernet 1/2/2
Device(config-if-e10000-1/2/2)# ipv6 address a000:1111::1/64
Device(config-if-e10000-1/2/2)# ipv6 pim-sparse
```

# ipv6 pimsm-snooping

Enables PIM6 SM traffic snooping.

## Syntax

**ipv6 pimsm-snooping**

**no ipv6 pimsm-snooping**

## Command Default

PIM6 SM traffic snooping is disabled.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

The device must be in multicast listening discovery (MLD) passive mode before it can be configured for PIM6 SM snooping.

Use PIM6 SM snooping only in topologies where multiple PIM sparse routers connect through a device. PIM6 SM snooping does not work on a PIM dense mode router which does not send join messages and traffic to PIM dense ports is stopped. A PIM6 SM snooping-enabled device displays a warning if it receives PIM dense join or prune messages.

When PIM6 SM snooping is enabled globally, you can override the global setting and disable it for a specific VLAN.

The **no** form of this command disables PIM6 SM traffic snooping.

## Examples

The following example enables PIM6 SM traffic snooping.

```
device(config)# ipv6 multicast passive
device(config)# ipv6 pimsm-snooping
```

The following example disables PIM6 SM traffic snooping.

```
device(config)# no ipv6 pimsm-snooping
```

The following example enables PIM6 SM traffic snooping on VLAN 20.

```
device(config)# vlan 20
device(config-vlan-20)# untagged ethernet 1/1/5 ethernet 1/1/7 ethernet 1/1/11
device(config-vlan-20)# multicast6 passive
device(config-vlan-20)# multicast6 pimsm-snooping
```

# ipv6 policy route-map

Enables IPv6 policy-based routing (PBR).

## Syntax

**ipv6 policy route-map** *route-map-name*

**no ipv6 policy route-map** *route-map-name*

## Command Default

PBR is not enabled.

## Parameters

*route-map-name*
Specifies the name of the route map.

## Modes

Global configuration mode

Interface configuration mode

Virtual interface configuration mode

## Usage Guidelines

This command can be used to enable IPv6 PBR globally on all interfaces or on a specific interface.

The **no** form of the command disables IPv6 PBR.

## Examples

The following example configures a route-map named test-route and enables IPv6 PBR globally.

```
device# configure terminal
device(config)# ipv6 access-list acl8 permit 2001:DB8:12d:1300::/64
device(config)# route-map test-route permit acl8
device(config-routemap test-route)# match ipv6 address acl8
device(config-routemap test-route)# set ipv6 next-hop 2001:DB8:12d:1300:1
device(config-routemap test-route)# exit
device(config)# ipv6 policy route-map test-route
```

The following example configures a route-map named test-route and enables IPv6 PBR on Ethernet interface 1/1/1.

```
device# configure terminal
device(config)# ipv6 access-list acl8 permit 2001:DB8:12d:1300::/64
device(config)# route-map test-route permit acl8
device(config-routemap test-route)# match ipv6 address acl8
device(config-routemap test-route)# set ipv6 next-hop 2001:DB8:12d:1300:1
device(config-routemap test-route)# exit
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ipv6 policy route-map map1
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# ipv6 prefix-list

Configures IPv6 prefix lists for use in basic traffic filtering.

## Syntax

**ipv6 prefix-list** { *name* | *sequence-number* } **deny** *ipv6-prefix/prefix-length* [ **ge** *ge-value* ] [ **le** *le-value* ]

**ipv6 prefix-list** { *name* | *sequence-number* } **description** *string*

**ipv6 prefix-list** { *name* | *sequence-number* } **permit** *ipv6-prefix/prefix-length* [ **ge** *ge-value* ] [ **le** *le-value* ]

**ipv6 prefix-list** { *name* | *sequence-number* } **seq** *sequence-number* **permit** *ipv6-prefix/prefix-length* [ **ge** *ge-value* ] [ **le** *le-value* ]

**ipv6 prefix-list** { *name* | *sequence-number* } **seq** *sequence-number* **deny** *ipv6-prefix/prefix-length* [ **ge** *ge-value* ] [ **le** *le-value* ]

**no ipv6 prefix-list** *name*

## Command Default

A prefix-list is not created.

## Parameters

*name*
Specifies the prefix list name.

*sequence-number*
Specifies an IPv6 prefix list sequence number.

**deny** *ip-prefix/prefix-length*
Denies a packet that contains a route specified in the prefix list. The prefix list matches only on the specified prefix/prefix length, unless you use the **ge** *ge-value* or **le** *le-value* parameters.

**ge** *ge-value*
Specifies a minimum range of prefix lengths, from *ge-value* to 128.

**le** *le-value*
Specifies a maximum range of prefix lengths, up to 128, from the *le-value* to the *prefix-length* parameter.

**description** *string*
Specifies a text string describing the prefix list.

**permit** *ip-prefix/prefix-length*
Permits a packet that contains a route specified in the prefix list. The prefix list matches only on the specified prefix/prefix length, unless you use the **ge** *ge-value* or **le** *le-value* parameters.

**seq** *sequence-number*
Specifies an IPv6 prefix list sequence number. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with prefix list entry 5. The device interprets the prefix list entries in numerical order, beginning with the lowest sequence number.

# Modes

Global configuration mode

# Usage Guidelines

An IPv6 prefix list is composed of one or more conditional statements that execute a permit or deny action if a packet matches a specified prefix. In prefix lists with multiple statements, you can specify a sequence number for each statement. The specified sequence number determines the order in which the statement appears in the prefix.

You can configure an IPv6 prefix list on a global basis, then use it as input to other commands or processes, such as route aggregation, route redistribution, route distribution, route maps, and so on. When a device interface sends or receives an IPv6 packet, it applies the statements within the IPv6 prefix list in their order of appearance to the packet. As soon as a match occurs, the device takes the specified action (permit or deny the packet) and stops further comparison for that packet.

You can use permit statements in the prefix list to specify the traffic that you want to send to the other feature. If you use deny statements, the traffic specified by the deny statements is not supplied to the other feature. You can configure up to one hundred IPv6 prefix lists.

You must specify the ipv6-prefix parameter in hexadecimal using 16-bit values between colons as documented in RFC 4291. You must specify the prefix-length parameter as a decimal value. A slash mark (/) must follow the ipv6-prefix parameter and precede the prefix-length parameter.

The *ge-value* or *le-value* you specify must meet the following condition for *prefix-length*:

```
ge-value <= le-value <= 128
```

If you do not specify **ge** *ge-value* or **le** *le-value*, the prefix list matches only on the exact prefix you specify with the *ipv6-prefix/prefix-length* parameter.

Prefix lists can be applied to RIPng globally using the separate **prefix-list** command) or at the interface level using the separate **ipv6 rip prefix-list** command. If both global IPv6 RIP prefix list and interface IPv6 rip prefix list are enabled, routes are filtered based on the interface prefix list.

The **no** form of the command deletes a prefix list.

# Examples

The following example creates a prefix-list that allows routes with the prefix 2001:db8::/32 to be included in RIPng routing updates sent from Ethernet interface 3/1/1.

```
device# configure terminal
device(config)# ipv6 prefix-list routesfor2001 permit 2001:db8::/32
device(config) # interface ethernet 3/1/1
device(config-if-e1000-3/1/1)# ipv6 rip prefix-list routesfor2001 out
```

The following example creates a prefix-list that allows routes with the prefix 2001:db8::/32 to be included in RIPng routing updates sent from all the IPv6 RIP interfaces on the device.

```
device# configure terminal
device(config)# ipv6 prefix-list routesfor2001 permit 2001:db8::/32
device(config)# ipv6 router rip
device(config-ripng-router)# distribute-list prefix-list routesfor2001 out
```

# ipv6-proto

Configures an IPv6 protocol-based VLAN.

## Syntax

**ipv6-proto** [ **name** *string* ]

**no ipv6-proto** [ **name** *string* ]

## Command Default

An IPv6 protocol-based VLAN is not configured.

## Parameters

**name** *string*

Specifies the IPv6 protocol-based VLAN name. The maximum length of the string is 32 characters.

## Modes

VLAN configuration mode

## Usage Guidelines

You can configure a protocol-based VLAN as a broadcast domain for IPv6 traffic. When the Layer 3 switch receives an IPv6 multicast packet (a packet with 06 in the version field and 0xFF as the beginning of the destination address), the Layer 3 switch forwards the packet to all other ports.

The **no** form of the command disables the IPv6 protocol VLAN.

## Examples

The following example configures the IPv6 protocol-based VLAN.

```
device(config)# vlan 2
device(config-vlan-2)# ipv6-proto name V6
```

# ipv6 raguard policy

Configures the specified Router Advertisement (RA) guard policy and enters RA guard policy configuration mode.

## Syntax

**ipv6 raguard policy** *name*

**no ipv6 raguard policy** *name*

## Parameters

*name*

An ASCII string indicating the name of the RA guard policy to configure.

## Modes

Global configuration mode

RA guard policy configuration mode

## Usage Guidelines

You can configure up to 256 RA guard policies.

The **no** form of this command deletes the specified RA guard policy.

## Examples

The following example configures an RA guard policy and enters RA guard policy configuration mode:

```
device(config)# ipv6 raguard policy policy1
device(ipv6-RAG-policy policy1)#
```

# ipv6 raguard vlan

Associates a Router Advertisement (RA) guard policy with a VLAN.

## Syntax

**ipv6 raguard vlan** *vlan-number* **policy** *name*

**no ipv6 raguard vlan** *vlan-number* **policy** *name*

## Parameters

*vlan-number*

Configures the ID number of the VLAN to which the specified RA guard policy should be associated. Valid range is from 1 to 4095.

**policy**

Associates a RA guard policy to the VLAN.

*name*

Specifies the name of the RA guard policy to be associated with the VLAN.

## Modes

Global configuration mode

## Usage Guidelines

A VLAN can have only one association with a RA guard policy. If you try to associate a new RA guard policy with a VLAN that is already associated with a policy, the new RA guard policy replaces the old one.

The **no** form of the command deletes the association of a RA guard policy from the VLAN.

## Examples

The following example associates RA guard policy named p1 with VLAN 1:

```
device(config)# ipv6 raguard vlan 1 policy p1
```

# ipv6 raguard whitelist

Configures the Router Advertisement (RA) guard whitelist and adds the IPv6 address as the allowed source IP address.

## Syntax

**ipv6 raguard whitelist** *whitelist-number* **permit** *ipv6-address*

**no ipv6 raguard whitelist** *whitelist-number* **permit** *ipv6-address*

## Parameters

*whitelist-number*
Configures the unique identifier for the RA guard whitelist. Valid values are 0 to 255.

**permit**
Configures the specified IPv6 address as the allowed source IP address to the RA guard whitelist.

*ipv6-address*
Configures the source IPv6 address. The address should be in the format X:X::X:X or X:X::X:X/M.

## Modes

Global configuration mode

## Usage Guidelines

You can configure source IP addresses from which RAs are permitted.

You can configure up to 64 RA guard whitelists, and each whitelist can have a maximum of 128 entries.

To remove the RA guard whitelist, use the **no** form the command without the **permit** keyword.

To remove a particular IPv6 address from the whitelist, use the **no** form of the command with the **permit***ipv6-address* keyword-variable pair.

When a whitelist associated with an RA guard policy is removed, all the entries in the whitelist are also removed. All the RAs are dropped because there is no whitelist associated with the RA guard policy.

## Examples

The following example configures an RA guard whitelist with the allowed source IP address:

```
device(config)# ipv6 raguard whitelist 1 permit fe80:db8::db8:10
```

The following example removes an RA guard whitelist:

```
device(config)# no ipv6 raguard whitelist 1
```

The following example removes a particular IPv6 address from the RA guard whitelist:

```
device(config)# no ipv6 raguard whitelist 1 permit fe80:db8::db8:10
```

# ipv6 redirects

Enables a Layer 3 switch to send an IPv6 ICMP redirect message to a neighboring host to inform it of a better first-hop router on a path to a destination.

## Syntax

**ipv6 redirects**

**no ipv6 redirects**

## Command Default

By default, the sending of IPv6 ICMP redirect messages by a Layer 3 switch is disabled.

## Modes

Interface configuration mode

## Usage Guidelines

This feature is supported on Virtual Ethernet (VE) interfaces only.

The **no** form of the command disables a Layer 3 switch to send an IPv6 ICMP redirect message to a neighboring host.

## Examples

The following example enables a Layer 3 switch to send an IPv6 ICMP redirect message to a neighboring host.

```
device(config)# interface ve 1
device(config-vif-1)# ipv6 redirects
```

# ipv6 rip default-information

Configures learning and advertising of default routes for RIPng.

## Syntax

**ipv6 rip default-information** { **only** | **originate** }

**no ipv6 rip default-information** { **only** | **originate** }

## Command Default

By default, the device does not learn IPv6 default routes.

## Parameters

**only**

Originates the default routes and suppresses all other routes from RIPng route updates.

**originate**

Originates the default routes and includes all other routes in the RIPng route updates.

## Modes

Interface configuration mode.

## Usage Guidelines

Use the **no** form of the command to remove the explicit default routes from RIPng and to suppress advertisement of these routes.

## Examples

The following example originates IPv6 default routes and includes all other routes in RIPng route updates sent from Ethernet interface 3/1/1.

```
device# configure terminal
device(config)# interface ethernet 3/1/1
device(config-if-e10000-3/1/1)# ipv6 rip default-information originate
```

# ipv6 rip enable

Enables RIPng on an interface.

## Syntax

**ipv6 rip enable**

**no ipv6 rip enable**

## Command Default

RIPng is disabled by default.

## Modes

RIPng configuration mode.

## Usage Guidelines

Use the **no** form of the command to disable RIPng on an individual interface.

Before you can enable RIPng, you must first enable forwarding of IPv6 traffic on the device using the **ipv6 unicast-routing** command. You must also enable IPv6 on each interface that will support RIPng. Enable IPv6 explicitly on an interface with the **ipv6 enable** command or by configuring an IPv6 address on the interface.

After you enable RIPng on the device using the **ipv6 router rip** command, use the **ipv6 rip enable** command to enable each RIPng interface individually. You can use the command to enable RIPng on a physical or virtual routing interface.

## Examples

The following example enables RIPng on Ethernet interface 3/1/1.

```
device# configure terminal
device(config)# interface ethernet 3/1/1
device(config-if-e100-3/1/1)# ipv6 rip enable
```

The following example enables RIPng on virtual ethernet interface 3.

```
device# configure terminal
device(config)# interface ve 3
device(config-vif-3)# ipv6 rip enable
```

# ipv6 rip metric-offset

Changes the metric for RIPng routes learned and advertised on an interface.

## Syntax

**ipv6 rip metric-offset** *value*

**ipv6 rip metric-offset out** *value*

**no ipv6 rip metric-offset** *value*

**no ipv6 rip metric-offset out** *value*

## Command Default

By default, an IPv6 RIP interface adds 1 to the metric of an incoming RIPng route that it learns. By default, the interface advertises RIPng routes without adding to the metric (that is, with a default offset of zero).

## Parameters

**out**

Specifies that the metric offset applies to outgoing (advertised) RIPng routes.

*value*

A decimal value that represents the offset to be added. The range is 1 through 16 for incoming routes and 0 through 15 for outgoing routes.

## Modes

Interface configuration mode.

## Usage Guidelines

Use the **no** form of these commands to return the metric offset to its default value, that is, 1 for incoming (learned) routes and 0 for outgoing (advertised) routes.

## Examples

The following example increases the metric on learned RIPng routes by 2. The same interface increases the metric offset by 3 when it advertises a RIPng route.

```
device# configure terminal
device(config)# interface ethernet 3/1/1
device(config-if-e1000-3/1/1)# ipv6 rip metric-offset 2
device(config-if-e1000-3/1/1)# ipv6 rip metric-offset out 3
```

# ipv6 rip summary-address

Advertises a summary of IPv6 addresses from an interface and specifies an IPv6 prefix that summarizes the routes.

## Syntax

**ipv6 rip summary-address** {*ipv6-prefix*/*prefix-length* }

**no ipv6 rip summary-address**{*ipv6-prefix*/*prefix-length* }

## Command Default

By default, original full-length routes rather than summary routes are advertised.

## Parameters

*ipv6-prefix*

Specifies the summarized IPv6 prefix as a hexadecimal value broken into 16-bit values separated by colons per RFC 2373.

*prefix-length*

Specifies the IPv6 prefix length in bits as a decimal value.

## Modes

Interface configuration mode.

## Usage Guidelines

Use the **no** form of the command to stop advertising the summarized IPv6 prefix.

The IPv6 prefix value must be separated from the prefix length by a forward slash ( / ).

## Examples

The following example advertises the summarized prefix 2001:db8::/36 instead of the IPv6 address 2001:db8:0:adff:8935:e838:78:e0ff/64 from Ethernet interface 3/1/1.

```
device# configure terminal
device(config)# interface ethernet 3/1/1
device(config-if-e40000-3/1/1)# ipv6 address 2001:db8:0:adff:8935:e838:78:
e0ff /64
device(config-if-e40000-3/1/1)# ipv6 rip summary-address 2001:db8::/36
```

# ipv6 route

Configures a static route.

## Syntax

**ipv6 route** [ **vrf** *vrf-name* ] *dest-ipv6-prefix* [ **ethernet** *unit/slot/port* | **ve** *ve-num* ] *next-hop-ipv6-address* [ *metric* ] [**distance** *number* ]

**no ipv6 route** [ **vrf** *vrf-name* ] *dest-ipv6-prefix* [ **ethernet** *unit/slot/port* | **ve** *ve-num* ] *next-hop-ipv6-address* [ *metric* ] [**distance** *number* ]

**ipv6 route** [ **vrf** *vrf-name* ] *dest-ipv6-prefix* { **tunnel** *num* | **null0** } [ *metric* ] [**distance** *number* ]

**no ipv6 route** [ **vrf** *vrf-name* ] *dest-ipv6-prefix* { **tunnel** *num* | **null0** } [ *metric* ] [**distance** *number* ]

## Parameters

**vrf** *vrf-name*
Specifies the VRF that contains the next-hop router (gateway) for the route.

*dest-ipv6-prefix*
Specifies the destination IPv6 address, including prefix length.

**ethernet** *unit/slot/port*
Configures the outgoing interface as the specified Ethernet interface.

**ve** *ve-num*
Configures the outgoing interface as the specified Virtual Ethernet interface.

*next-hop-ipv6-address*
Specifies the IPv6 address of a next-hop gateway. The next-hop address may be a global IPv6 address or a link-local IPv6 address.

*metric*
Specifies the route's metric. The value can range from 1 to 16. The default value is 1.

**distance** *number*
Specifies the route's administrative distance. The default value is 1.

**tunnel** *num*
Configures the outgoing interface as the specified tunnel interface.

**null0**
Drops packets with this destination.

## Modes

Global configuration mode

## Usage Guidelines

By default, static routes take precedence over routes learned by routing protocols.

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and must enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

ICX 7150 devices do not support tunnels or VRFs.

If a non-default VRF is configured, the **no** form of the command removes the static IPv6 route configuration from a VRF. When no VRF is configured, the **no** form of the command removes the IPv6 static route.

When a tunnel is configured as the next hop for a static route, the tunnel must already be configured if the destination is a non-default VRF. In contrast, a tunnel can be designated as the next hop in the default VRF before it is configured. The default VRF is used when no VRF is specified in the command.

# Examples

The following example configures a static IPv6 route for a destination network with the prefix 2001:DB8::0/32 and a next-hop gateway with the global address 2001:DB8:0:ee44::1.

```
device(config)# ipv6 route 2001:DB8::0/32 2001:DB8:0:ee44::1
```

The following example configures a static IPv6 route for a destination network with the prefix 2001:DB8::0/32 and a next-hop gateway with the global address 2001:DB8:0:ee44::1 in the non-default VRF named blue.

```
device(config)# ipv6 route vrf blue 2001:DB8::0/32 2001:DB8:0:ee44::1
```

The following example configures tunnel 1 as the next hop gateway for 2001:DB8::0/32 destination addresses. Because the destination is a non-default VRF (VRF blue), the tunnel must be configured before the static route is configured.

```
device(config)# ipv6 route vrf blue 2001:DB8::0/32 tunnel 1
```

The following example configures a null route that discards packets to the destination ipv6 route 2001 : DB8 : : 0/32 when the preferred route using virtual interface 3 (ve 3) through the next hop with the link-local address fe80::1 is not available. The null route has a higher metric (2) than the preferred route, which has a default metric of 1.

```
device# configure terminal
device(config)# ipv6 route 2001 : DB8 : : 0/32 ve 3 fe80::1
device(config)# ipv6 route 2001 : DB8 : : 0/32 null0 2
```

# ipv6 route next-hop

You can resolve IPv6 static routes through the specified protocol.

## Syntax

**ipv6 route** [ **vrf** *vrf-name* ] **next-hop** { **bgp** | **ospf** | **rip** }

**no ipv6 route** [ **vrf** *vrf-name* ] **next-hop** { **bgp** | **ospf** | **rip** }

## Command Default

By default, static routes are not distributed or resolved through another protocol.

## Parameters

**vrf** *vrf-name*
Specifies the VRF that contains the next-hop router (gateway) for the route.

**next-hop** { **bgp** | **ospf** | **rip** }
Specifies a protocol (BGP, OSPF, or RIP) to be used to resolve the IPv6 static route.

## Modes

Global configuration mode

## Usage Guidelines

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

ICX 7150 devices do not support tunnels or VRFs.

The **no** form of the command disables IPv6 static route next-hop resolution through the designated protocol. If a VRF is configured, the **no** form of the command removes the static IPv6 route configuration from the VRF.

## Examples

The following example enables IPv6 static route next-hop resolution through OSPF.

```
device# configure terminal
device(config)# ipv6 route next-hop ospf
```

# ipv6 route next-hop-enable-default

You can enable the IPv6 default static route to resolve other static routes.

## Syntax

**ipv6 route** [ **vrf** *vrf-name* ] **next-hop-enable-default**

**no ipv6 route** [ **vrf** *vrf-name* ] **next-hop-enable-default**

## Command Default

By default, the IPv6 default static route is not used to resolve static route next hops.

## Parameters

**vrf** *vrf-name*

Specifies the VRF that contains the next-hop router (gateway) for the route.

## Modes

Global configuration mode

## Usage Guidelines

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and must enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

ICX 7150 devices do not support tunnels or VRFs.

The **no** form of the command disables IPv6 static route next-hop resolution through the default route. If a VRF is configured, the **no** form of the command removes the static IPv6 route configuration from the VRF.

## Examples

The following example configures static routing next-hop recursion to three levels (the default). It configures the network default static route to global IPv6 address 2001:DB8:0:ee44::1 and allows it to resolve other static routes.

**NOTE**

You can specify a level of recursion up to 10.

```
device# configure terminal
device(config)# ipv6 route next-hop-recursion
device(config)# ipv6 route ipv6 route ::/0 2001:DB8:0:ee44::1
device(config)# ipv6 route next-hop-enable-default
```

# ipv6 route next-hop-recursion

You can resolve static route destination using recursive lookup in local address tables up to 10 hops away.

## Syntax

**ipv6 route** [ **vrf** *vrf-name* ] **next-hop-recursion** [ *number* ]

**no ipv6 route** [ **vrf** *vrf-name* ] **next-hop-recursion** [ *number* ]

## Command Default

By default, only the local IPv6 address table is consulted to resolve the next hop toward a static route destination.

## Parameters

**vrf** *vrf-name*

Specifies the VRF that contains the next-hop router (gateway) for the route.

*number*

Specifies the level of recursion for address lookup. The range is 1 through 10. If no number is specified, the default value is 3.

## Modes

Global configuration mode

## Usage Guidelines

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and must enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

ICX 7150 devices do not support tunnels or VRFs.

The **no** form of the command disables IPv6 static route next-hop recursion. If a VRF is configured, the **no** form of the command removes the static IPv6 route configuration from the VRF.

## Examples

The following example configures recursive static route lookup to five levels for static route resolution.

```
device# configure terminal
device(config)# ipv6 route next-hop-recursion 5
```

# ipv6 router ospf

Enables and configures the Open Shortest Path First version 3 (OSPFv3) routing protocol.

## Syntax

**ipv6 router ospf** [ **vrf** *name* ]

**no ipv6 router ospf**

## Command Default

Disabled.

## Parameters

**vrf** *name*

Specifies a nondefault VRF.

## Modes

Global configuration mode

## Usage Guidelines

If you save the configuration to the startup-config file after disabling OSPFv3, all OSPFv3 configuration information is removed from the startup-config file.

Use this command to enable the OSPFv3 routing protocol and enter OSPFv3 router or OSPFv3 router VRFconfiguration mode. OSPFv3 maintains multiple instances of the routing protocol to exchange route information among various VRF instances.

The **no** form of the command deletes all current OSPFv3 configurations and blocks any further OSPFv3 configuration.

## Examples

The following example enables OSPFv3 on a default VRF and enters OSPFv3 router configuration mode.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)#
```

# ipv6 router pim

Enables IPv6 PIM-Sparse mode for IPv6 routing globally or on a specified VRF.

## Syntax

**ipv6 router pim** [ **vrf** *vrf-name* ]

**no ipv6 router pim** [ **vrf** *vrf-name* ]

## Command Default

IPv6 PIM-Sparse mode is not enabled.

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

## Modes

Global configuration mode.

VRF configuration mode.

## Usage Guidelines

The **no** form of this command removes the IPv6 PIM-Sparse mode configuration.

## Examples

The following example enables IPv6 PIM-Sparse mode on a VRF named blue.

```
Device(config)# ipv6 router pim vrf blue
```

# ipv6 router rip

Enables RIPng globally (on the device).

## Syntax

**ipv6 router rip**

**no ipv6 router rip**

## Command Default

By default, RIPng is disabled.

## Modes

Global configuration mode

## Usage Guidelines

To disable RIPng globally, use the **no** form of this command.

Before you can enable RIPng, you must enable forwarding of IPv6 traffic on the device using the **ipv6 unicast-routing** command.

You must enable IPv6 on each interface on which RIPng is to be enabled. Enable IPv6 explicitly on the interface with the **ipv6 enable** command or by configuring an IPv6 address on the interface.

After enabling RIPng globally, you must enable it on individual device interfaces using the **ipv6 rip enable** command. You can enable RIPng on physical as well as virtual routing interfaces.

## Examples

The following example enables RIPng on the device.

```
device# configure terminal
device(config)# ipv6 router rip
device(config-ripng-router)#
```

# ipv6 router vrrp

Globally enables IPv6 Virtual Router Redundancy Protocol (VRRP).

## Syntax

**ipv6 router vrrp**

**no ipv6 router vrrp**

## Command Default

IPv6 VRRP is not globally enabled.

## Modes

Global configuration mode

## Usage Guidelines

After globally enabling IPv6 VRRP, the command prompt does not change. Nearly all subsequent IPv6 VRRP configuration is performed at the interface level, but IPv6 VRRP must be enabled globally before configuring IPv6 VRRP instances.

The **no** form of the command disables VRRP globally.

**NOTE**
Only 16 VRRP instances are configurable on the ICX 7150 device.

## Examples

The following example enables IPv6 VRRP globally and enters interface configuration mode to allow you to enter more VRRP configuration.

```
device# configure terminal
device(config)# ipv6 router vrrp
device(config-ipv6-vrrp-router)# interface ethernet 1/1/4
device(config-if-e1000-1/1/4)# ipv6 address fd3b::3/64
device(config-if-e1000-1/1/4)# ipv6 vrrp vrid 2
device(config-if-e1000-1/1/4-vrid-2)# backup priority 100
device(config-if-e1000-1/1/4-vrid-2)# version 3
device(config-if-e1000-1/1/4-vrid-2)# advertise backup
device(config-if-e1000-1/1/4-vrid-2)# ipv6-address fe80::768e:f8ff:fe2a:0099
device(config-if-e1000-1/1/4-vrid-2)# ipv6-address fd3b::2
device(config-if-e1000-1/1/4-vrid-2)# activate
```

# ipv6 router vrrp-extended

Globally enables IPv6 Virtual Router Redundancy Protocol Extended (VRRP-E).

## Syntax

**ipv6 router vrrp-extended**

**no ipv6 router vrrp-extended**

## Command Default

VRRP-E is not globally enabled.

## Modes

Global configuration mode

## Usage Guidelines

After globally enabling IPv6 VRRP-E, nearly all subsequent IPv6 VRRP-E configuration is performed at the interface level. If IPv6 VRRP-E is not globally enabled, you will see an error message when configuring IPv6 VRRP-E instances.

The **no** form of the command disables VRRP-E globally.

**NOTE**
Only 16 VRRP instances are configurable on the ICX 7150 device.

## Examples

The following example enables IPv6 VRRP-E globally and enters interface configuration mode for subsequent IPv6 VRRP-E configuration.

```
device# configure terminal
device(config)# ipv6 router vrrp-extended
device(config-ipv6-vrrpe-router)# interface ethernet 1/1/5
```

# ipv6 traffic-filter

Applies an IPv6 access control list (ACL) to incoming or outgoing traffic on an interface.

## Syntax

**ipv6 traffic-filter** *acl-name* { **in** | **out** }

**no ipv6 traffic-filter** *acl-name* { **in** | **out** }

## Command Default

The ACL is not applied to an interface.

## Parameters

*acl-name*

Applies the specified ACL to interface traffic.

**in**

Applies the specified IPv6 ACL to incoming IPv6 packets on the interface.

**out**

Applies the specified IPv6 ACL to outgoing IPv6 packets on the interface.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command removes the ACL from interface traffic.

## Examples

The following example applies the ACL "acl1" to inbound traffic on Ethernet interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ipv6 traffic-filter acl1 in
```

# ipv6 unicast-routing

Enables the forwarding of IPv6 traffic on a Layer 3 swtich.

## Syntax

**ipv6 unicast-routing**

**no ipv6 unicast-routing**

## Command Default

The forwarding of IPv6 traffic is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

Before configuring a static IPv6 route, you must enable the forwarding of IPv6 traffic on the Layer 3 switch using the **ipv6 unicast-routing** command and must enable IPv6 on at least one interface by configuring an IPv6 address or explicitly enabling IPv6 on that interface.

The **no** form of the command disables IPv6 unicast routing.

## Examples

The following example enables IPv6 unicast routing.

```
device(config)# ipv6 unicast-routing
```

# ipv6 vrrp vrid

Configures an IPv6 Virtual Router Redundancy Protocol (VRRP) virtual router identifier (VRID).

## Syntax

**ipv6 vrrp vrid** *vrid*

**no ipv6 vrrp vrid** *vrid*

## Command Default

An IPv6 VRRP VRID does not exist.

## Parameters

*vrid*

Configures a number for the IPv6 VRRP VRID. The range is from 1 through 255.

## Modes

Interface configuration mode

## Usage Guidelines

Before configuring this command, ensure that IPv6 VRRP is enabled globally; otherwise, an error stating “Invalid input...” is displayed as you try to create a VRRP instance.

The **no** form of this command removes the IPv6 VRRP VRID from the configuration.

## Examples

The following example configures IPv6 VRRP VRID 1.

```
device# configure terminal
device(config)# ipv6 router vrrp
device(config)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ipv6 address fd2b::2/64
device(config-if-e1000-1/1/5)# ipv6 vrrp vrid 2
device(config-if-e1000-1/1/5-vrid-2)# owner
device(config-if-e1000-1/1/5-vrid-2)# ipv6-address fe80::768e:f8ff:fe2a:0099
device(config-if-e1000-1/1/5-vrid-2)# ipv6-address fd2b::2
device(config-if-e1000-1/1/5-vrid-2)# activate
```

# ipv6 vrrp-extended vrid

Configures an IPv6 Virtual Router Redundancy Protocol Extended (VRRP-E) virtual router identifier (VRID).

## Syntax

**ipv6 vrrp-extended vrid** *vrid*

**no ipv6 vrrp-extended vrid** *vrid*

## Command Default

An IPv6 VRRP-E VRID does not exist.

## Parameters

*vrid*

Configures a number for the IPv6 VRRP-E VRID. The range is from 1 through 255.

## Modes

Interface configuration mode

## Usage Guidelines

Before configuring this command, ensure that IPv6 VRRP-E is enabled globally; otherwise, an error stating “Invalid input...” is displayed as you try to create a VRRP-E instance.

The **no** form of this command removes the IPv6 VRRP-E VRID from the configuration.

## Examples

The following example configures IPv6 VRRP-E VRID 2.

```
device# configure terminal
device(config)# ipv6 router vrrp-extended
device(config-ipv6-vrrpe-router)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ipv6 address fd4b::2/64
device(config-if-e1000-1/1/5)# ipv6 vrrp-extended vrid 2
device(config-if-e1000-1/1/5-vrid-2)# backup  priority 50 track-priority 10
device(config-if-e1000-1/1/5-vrid-2)# ipv6-address fe80::768e:f8ff:fe3a:0099
device(config-if-e1000-1/1/5-vrid-2)# ipv6-address fd4b::99
device(config-if-e1000-1/1/5-vrid-2)# activate
```

# ipv6-address

Configures a virtual IPv6 address for a Virtual Router Redundancy Protocol version 3 (VRRPv3) or VRRP Extended version 3 (VRRP-Ev3) instance.

## Syntax

**ipv6-address** { *ipv6-address* | **auto-gen-link-local** }

**no ipv6-address** { *ipv6-address* | **auto-gen-link-local** }

## Command Default

A virtual IPv6 address is not configured for a VRRPv3 or VRRP-Ev3 instance.

## Parameters

*ipv6-address*
: Configures an IPv6 address.

**auto-gen-link-local**
: Automatically generates a virtual IPv6 link-local address for the VRRPv3 instance. Not supported in VRRP-Ev3.

## Modes

Virtual routing ID interface configuration mode

## Usage Guidelines

For VRRP instances, the IPv6 address used for the virtual router must be configured on the device assigned to be the initial VRRP owner device. The same physical IPv6 address cannot be used on any other VRRP device.

If the **auto-gen-link-local** keyword is entered, a virtual IPv6 link-local address is generated automatically for the specific VRRPv3 instance. The virtual link-local address is carried in VRRPv3 advertisements. A manually configured link-local address takes precedence over the automatically generated address.

**NOTE**
Automatically generated virtual link-local addresses are not supported for VRRP-Ev3 instances.

The **no** form of the command removes the virtual router IPv6 address. If the **auto-gen-link-local** keyword was active, the automatically generated virtual IPv6 link-local address is removed for the VRRPv3 instance, and subsequent VRRPv3 advertisements will not carry this link-local address.

# Examples

The following example configures a virtual IPv6 address for VRID 1 when IPv6 VRRPv3 is implemented. In this example, the device is configured as the VRRPv3 owner device.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ipv6 address fd2b::1/64
device(config-if-e1000-1/1/6)# ipv6 vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner
device(config-if-e1000-1/1/6-vrid-1)# ipv6-address fe80::768e:f8ff:fe2a:0099
device(config-if-e1000-1/1/6-vrid-1)# ipv6-address fd2b::1
device(config-if-e1000-1/1/6-vrid-1)# activate
```

The following example configures a virtual IPv6 address for VRID 2 when VRRP-Ev3 is implemented. In this example, the device is configured as a VRRP-Ev3 backup device and the highest priority device will become the master VRRP-Ev3 device.

```
device# configure terminal
device(config)# ipv6 router vrrp-extended
device(config-ipv6-vrrpe-router)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ipv6 address fd4b::1/64
device(config-if-e1000-1/1/5)# ipv6 vrrp-extended vrid 2
device(config-if-e1000-1/1/5-vrid-2)# backup priority 110
device(config-if-e1000-1/1/5-vrid-2)# ipv6-address fe80::768e:f8ff:fe3a:0099
device(config-if-e1000-1/1/5-vrid-2)# ipv6-address fd4b::99
device(config-if-e1000-1/1/5-vrid-2)# activate
```

# ipv6-address auto-gen-link-local

Generates a virtual link-local IPv6 address and assigns it as the virtual IPv6 address for a VRRPv3 instance.

## Syntax

**ipv6-address auto-gen-link-local**

**no ipv6-address auto-gen-link-local**

## Modes

VRRP sub-configuration mode

## Usage Guidelines

The **no** form of this command deletes the auto-generated virtual link-local IPv6 address for the VRRP v3 instance.

The default VRRPv3 implementation allows only the link-local address that is configured on a physical interface to be used as the virtual IPv6 address of a VRRPv3 instance. This limits configuring a link-local address for each VRRP instance on the same physical interface because there can be only one link-local address per physical interface. You can use this command on the owner or backup router to generate a virtual link-local IPv6 address from the virtual MAC address of a VRRPv3 instance and assign it as the virtual IPv6 address for the VRRPv3 instance. This auto-generated link-local IPv6 address is not linked to any physical interface on the router.

## Examples

The following example generates a virtual link-local IPv6 address and its allocation as the virtual IPv6 address of a VRRPv3 cluster on an owner router.

```
device(config)# interface ve 3
device(config-vif-3)# ipv6 vrrp vrid 2
device(config-vif-3-vrid-2)# owner
device(config-vif-3-vrid-2)# ipv6-address auto-gen-link-local
device(config-vif-3-vrid-2)# activate
```

## History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced. |

# ipv6-neighbor inspection trust

Enables trust mode for specific ports.

## Syntax

**ipv6-neighbor inspection trust** [ **vrf** *vrf-name* ]

**no ipv6-neighbor inspection trust** [ **vrf** *vrf-name* ]

## Command Default

Trust mode is not enabled. When you enable ND inspection on a VLAN, by default, all the interfaces and member ports are considered as untrusted.

## Parameters

**vrf**

Specifies the VRF instance.

*vrf-name*

Specifies the ID of the VRF instance.

## Modes

Interface configuration mode

VRF configuration mode

## Usage Guidelines

The **no** form of the command disables trust mode on ports.

## Examples

The following example displays the trust mode configuration for ports.

```
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# ipv6-neighbor inspection trust
```

The following example displays the trust mode configuration on a port on VRF 3.

```
device(config-if-e1000-1/1/1)# ipv6-neighbor inspection trust vrf 3
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# ipx-network

Configures the IPX network protocol-based VLANs.

## Syntax

**ipx-network** *network-number ipx-frame-type* [ **name** *string* ]

**no ipx-network** *network-number ipx-frame-type* [ **name** *string* ]

## Command Default

An IPX network protocol-based VLAN is not configured.

## Parameters

*network-number*

Specifies the network number in hexadecimal format.

*ipx-frame-type*

Defines the IPX frame encapsulation standard types. The following are the supported encapsulation standard types:

**ethernet_802.2**

Specifies the Ethernet 802.2 standard that can be configured for the protocol.

**ethernet_802.3**

Specifies the Ethernet 803.3 standard that can be configured for the protocol.

**ethernet_ii**

Specifies the Ethernet II standard that can be configured for the protocol.

**ethernet_snap**

Specifies the Ethernet subnetwork access protocol standard that can be configured for the protocol.

**name** *string*

Specifies the Ethernet standard name. The string can be up to 32 characters in length.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command disables the IPX network protocol-based VLAN.

# Examples

The following example shows how to configure the IPX network protocol-based VLAN.

```
device(config)# vlan 20 name IPX_VLAN by port
device(config-vlan-10)# untagged ethernet 1/2/1 to 1/2/6
added untagged port ethe 1/2/1 to 1/2/6 to port-vlan 20.
device(config-vlan-10)# ipx-network abcd ethernet_ii name Eng-LAN
```

# ipx-proto

Configures the IPX protocol-based VLANs.

## Syntax

**ipx-proto** [ **name** *string* ]

**no ipx-proto** [ **name** *string* ]

## Command Default

An IPX protocol-based VLAN is not configured.

## Parameters

**name** *string*

The IPX protocol-based VLAN name. The name can be up to 32 characters in length.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the IPX protocol-based VLAN.

## Examples

The following example shows the how to configure an IPX protocol-based VLAN.

```
device(config)# vlan 10 by port
device(config-vlan-10)# untagged ethernet 1/1/1 to 1/1/6
added untagged port ethe 1/1/1 to 1/1/6 to port-vlan 30.
device(config-vlan-10)# ip-proto name IP_Prot_VLAN
```

# issu abort

Initiates an in service software upgrade (ISSU) termination.

## Syntax

**issu abort**

## Modes

Privileged EXEC mode

## Usage Guidelines

This is a command that an operator uses to manually stop the currently running upgrade.

The upgrade terminates after the current unit, the one that is going through the upgrade, rejoins the stack.

If a manual abort is done or ISSU detects an abort condition (with ISSU started with no **on-error** option), the stack is left as it is and a manual recovery is required. You must reload the primary or secondary image to bring the stack back to working condition after issuing the **issu abort** command.

## Examples

Follow this example to terminate an ISSU.

```
device# issu abort
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Adds Campus Fabric (SPX) system support. |

# issu primary

Initiates an in-service software upgrade (ISSU) using the image on the primary partition and configures the system to reload from either the primary image or the secondary image if the upgrade fails.

## Syntax

**issu primary** [ **on-error** { **reload-primary** | **reload-secondary** } ]

## Command Default

If an error occurs ths default behavior is to abort the ISSU.

## Parameters

**on-error**

Specifies the action to take if there is an upgrade failure from the primary image.

**reload-primary**

Causes the system to reload from the primary partition if an upgrade from the primary partition fails.

**reload-secondary**

Causes the system to reload from the secondary partition if an upgrade from the primary partition fails.

## Modes

Privileged EXEC mode

## Usage Guidelines

Before you use this command, back up the running image to the secondary partition, use the existing image upgrade framework to copy the new image to the primary or secondary partition, and check the sequence of the upgrade with the **show issu sequence** command.

The **issu primary** command without any keywords initiates an ISSU.

If a manual abort is done or ISSU detects an abort condition (with ISSU started with no **on-error** option), the stack is left as is and a manual recovery is required.

If a manual recovery is required, you run either the **reload-primary** or **reload-secondary** command.

# Examples

The following example shows how to start an ISSU, using an image that has been copied to the primary partition.

```
device# issu primary
Topology is Ring                        Yes
Standby Present                         Yes
Standby ready for upgrade               Yes
Flash use in progress                   No
Secure Setup in progress                No
ISSU in progress or aborted             No
Election pending                        No
Election in progress                    No
Reload pending                          No
CPU utilization high                    No
All units in ready state                Yes
Primary Image is upgrade compatible     Yes
Startup config and Running Config Same  Yes
User in Config mode                     No
Proceed with upgrade? (enter 'y' or 'n'):
```

If the system is not ready for an ISSU, the error condition is highlighted.

```
device# issu primary
Topology is Ring                        Yes
Standby Present                         No    ***
Standby ready for upgrade               No    ***
Flash use in progress                   No
Secure Setup in progress                No
ISSU in progress or aborted             No
Election pending                        No
Election in progress                    No
Reload pending                          No
CPU utilization high                    No
All units in ready state                Yes
Primary Image is upgrade compatible     Yes
Secondary Image is upgrade compatible   Yes
Startup config and Running Config Same  Yes
User in Config mode                     No
System not ready for issu. Check error condition highlighted by "***" and rectify.
ISSU not in progress
```

The behavior in case ISSU fails can be specified. In the following example the specified behavior is to reload the image on the secondary partition.

```
device# issu primary on-error reload-secondary
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Adds Campus Fabric (SPX) system support. |

# issu secondary

Initiates an in-service software upgrade (ISSU) using the image on the secondary partition and configures the system to reload from the image in either the primary partition or the secondary partition should the upgrade fail.

## Syntax

**issu secondary** [ **on-error** { **reload-primary** | **reload-secondary** } ]

## Command Default

If an error occurs ths default behavior is to abort the ISSU.

## Parameters

**on-error**

Specifies the action to take if there is an upgrade failure from the secondary image.

**reload-primary**

Causes the system to reload from the primary partition if an upgrade from secondary partition fails.

**reload-secondary**

Causes the system to reload from the secondary partition if an upgrade from secondary partition fails.

## Modes

Privileged EXEC mode

## Usage Guidelines

Before you use this command, back up the running image to the primary partition, use the existing image upgrade framework to copy the new image to the primary or secondary partition, and check the sequence of upgrade with the **show issu sequence** command.

The **issu secondary** command without any keywords initiates an ISSU.

If a manual abort is done or ISSU detects an abort condition (with ISSU started with no **on-error** option), the stack is left as is and a manual recovery is required.

If a manual recovery is required, you run either the **reload-primary** or **reload-secondary** command.

# Examples

The following example shows how to start an ISSU.

```
device# issu secondary
Topology is Ring                        Yes
Standby Present                         Yes
Standby ready for upgrade               Yes
Flash use in progress                   No
Secure Setup in progress                No
ISSU in progress or aborted             No
Election pending                        No
Election in progress                    No
Reload pending                          No
CPU utilization high                    No
All units in ready state                Yes
Primary Image is upgrade compatible     Yes
Startup config and Running Config Same  Yes
User in Config mode                     No
Proceed with upgrade? (enter 'y' or 'n'):
```

If the system is not ready for an ISSU, the error condition is highlighted.

```
device# issu secondary
Topology is Ring                        Yes
Standby Present                         No    ***
Standby ready for upgrade               No    ***
Flash use in progress                   No
Secure Setup in progress                No
ISSU in progress or aborted             No
Election pending                        No
Election in progress                    No
Reload pending                          No
CPU utilization high                    No
All units in ready state                Yes
Primary Image is upgrade compatible     Yes
Secondary Image is upgrade compatible   Yes
Startup config and Running Config Same  Yes
User in Config mode                     No
System not ready for issu. Check error condition highlighted by "***" and rectify.
ISSU not in progress
```

The behavior in case ISSU fails can be specified. In the following example the specified behavior is to reload the image on the secondary partition.

```
device# issu secondary on-error reload-secondary
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Adds Campus Fabric (SPX) system support. |

# Commands J, K, and L

## jitc enable

Enables the Joint Interoperability Test Command (JITC) mode.

### Syntax

**jitc enable**

**no jitc enable**

### Command Default

JITC is not enabled.

### Modes

Global configuration mode

### Usage Guidelines

When JITC is enabled, the Advanced Encryption Standard - Cipher-Block Chaining (AES-CBC) encryption mode for the Secure Shell (SSH) protocol is disabled and the AES-CTR (Counter) encryption mode is enabled.

When JITC is enabled, the MD5 authentication scheme for NTP is disabled.

The **no** form of the command disables the JITC mode and puts the system back to the standard mode and enables both AES-CBC encryption mode and MD5 authentication configuration.

### Examples

The following example enables the JITC mode.

```
device(config)# jitc enable
```

### History

| Release version | Command history |
|---|---|
| 08.0.20a | This command was introduced. |

# jitc show

Displays the status of the JITC mode.

## Syntax

**jitc show**

## Modes

Global configuration mode

Privileged EXEC mode

## Command Output

The **jitc show** command displays the following information.

| Output field | Description |
|---|---|
| JITC mode | Displays the status of the JITC mode. |
| SSH AES-CTR mode | Displays the status of the SSH AES-CTR mode. |
| SSH AES-CBC mode | Displays the status of the SSH AES-CBC mode. |

## Examples

The following example shows the output of the **jitc show** command.

```
device(config)#jitc show
JITC mode : Enabled
Management Protocol Specific:
SSH AES-CTR mode : Enabled
SSH AES-CBC mode : Disabled
```

## History

| Release version | Command history |
|---|---|
| 08.0.20a | This command was introduced. |

# join-timer leave-timer leaveall-timer

Changes the Join, Leave, and Leaveall timers for GVRP counters.

## Syntax

**join-timer** *join-timer-ms* **leave-timer** *leave-timer-ms* **leaveall-timer** *leaveall-timer-ms*

## Command Default

The default value for the Join timer is 200 ms. The default value for the Leave timer is 600 ms. The default value for the Leaveall timer is 10,000 ms.

## Parameters

*join-timer-ms*

Specifies the maximum number of milliseconds (ms) a GVRP interface wait before sending VLAN advertisements on the interfaces. You can set the Join timer to a value from 200 to one third the value of the Leave timer.

*leave-timer-ms*

Specifies the number of milliseconds a GVRP interface waits after receiving a Leave message on the port to remove the port from the VLAN indicated in the Leave message. You can set the Leave timer to a value from three times the Join timer value to one fifth the value of the Leaveall timer.

*leaveall-timer-ms*

Specifies the minimum interval at which GVRP sends Leaveall messages on all GVRP interfaces. You can set the Leaveall timer to a value from five times the Leave timer value to the maximum value allowed by the software (configurable from 300,00 to 1,000,000 ms).

## Modes

GVRP configuration mode

## Usage Guidelines

All timer values must be in multiples of 100 ms.

The Leave timer value must be greater than or equal to three times the Join timer value. The Leaveall timer value must be greater than or equal to five times the Leave timer value.

The GVRP timers must be set to the same values on all the devices that are exchanging information using GVRP.

**NOTE**
When you enter this command, all the running GVRP timers are canceled and restarted using the new times specified by the command.

# Examples

The following example shows how to set the Join, Leave, and Leaveall timers.

```
device(config)# gvrp-enable
device(config-gvrp)# join-timer 1000 leave-timer 3000 leaveall-timer 15000
```

# jumbo

Provides support for jumbo frames.

## Syntax

**jumbo**

**no jumbo**

## Command Default

Jumbo frame support is disabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables jumbo frame support.

## Examples

The following example provides jumbo frame support.

```
device(config)# jumbo
```

# keep-alive-vlan

Configures a keep-alive VLAN for the cluster.

## Syntax

**keep-alive-vlan** *vlan-ID*

**no keep-alive-vlan** *vlan-ID*

## Command Default

A keep-alive VLAN is not configured.

## Parameters

*vlan-ID*

Specifies the VLAN number. The values can be from 1 through 4089.

## Modes

Cluster configuration mode

## Usage Guidelines

Only one VLAN can be configured as a keep-alive VLAN. The keep-alive VLAN cannot be a member VLAN of the Multi-Chassis Trunking (MCT) and this VLAN can be tagged or untagged.

When the CCP is down, the following results occur:

- If the keep-alive VLAN is configured, CCRR messages are sent every second over that VLAN.
- If no packets are received from the peer device for a period of three seconds, the peer is considered down.
- If a keep-alive VLAN is not configured and both the peer devices are up, both peers continue forwarding traffic independently, when the CCP is down.

  **NOTE**
  Keep-alive VLAN configuration is not allowed when the client isolation mode is strict; and when the keep-alive VLAN is configured, client isolation mode cannot be configured as strict.

The **no** form of the command removes the keep-alive VLAN configuration.

## Examples

The following example shows how to configure the keep-alive VLAN.

```
device(config)# cluster SX 400
device(config-cluster-SX)# keep-alive-vlan 10
```

# keepalive

Configures GRE link keepalive.

## Syntax

**keepalive** [ *interval* [ *retries* ] ]

**no keepalive** [ *interval* [ *retries* ] ]

## Command Default

GRE link keepalive is not enabled.

## Parameters

*interval*

Specifies the number of seconds between each initiation of a keepalive message. The range if from 2 to 32767 seconds and the default is 10 seconds.

*retries*

Specifies the number of times that a packet is sent before the system places the tunnel in the DOWN state. Valid values are from 1 through 255. The default number of retries is 3.

## Modes

Tunnel interface configuration mode

## Usage Guidelines

When GRE tunnels are used in combination with static routing or policy-based routing, and a dynamic routing protocol such as RIP, BGP, or OSPF is not deployed over the GRE tunnel, a configured tunnel does not have the ability to bring down the line protocol of either tunnel endpoint, if the far end becomes unreachable. Traffic sent on the tunnel cannot follow alternate paths because the tunnel is always UP. To avoid this scenario, enable GRE link keepalive, which will maintain or place the tunnel in an UP or DOWN state based upon the periodic sending of keepalive packets and the monitoring of \responses to the packets. If the packets fail to reach the tunnel far end more frequently than the configured number of retries, the tunnel is placed in the DOWN state

The **no** form of the command disables GRE keepalive.

## Examples

The following example enables GRE keepalive and sets the keepalive interval and retries.

```
device(config)# keepalive 500 150
```

# keepalive (IKEv2)

Configures the interval between Internet Key Exchange version 2 (IKEv2) messages that are sent to detect if a peer is still alive.

## Syntax

**keepalive** *interval*

**no keepalive** *interval*

## Command Default

IKEv2 keepalive is enabled and the keepalive interval is 300 seconds.

## Parameters

*interval*

Specifies the number of seconds between each initiation of an IKEv2 notify message. The range is from 10 through 3600. A value of 0 disables the keepalive function.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

The **no** form of the command restores the default configuration.

## Examples

The following example shows how to configure a keepalive interval of 500 seconds for an IKEv2 profile named prof_mktg.

```
device(config)ikev2 profile prof_mktg
device(config-ike-profile-prof_mktg)# keepalive 500
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# keychain

Configures a name for the keychain.

## Syntax

**keychain** *keychain-name*

**no keychain** *keychain-name*

## Command Default

A keychain is not configured by default.

## Parameters

*keychain-name*
Specifies the name of the keychain.

## Modes

Global configuration mode

## Usage Guidelines

A maximum of up to 64 keychains can be configured.

This command takes the configuration to the keychain configuration mode, in which key identifiers can be added.

The **no** form of the command removes the keychain.

## Examples

The following example configures a keychain.

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)#
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# key-id

Configures a key in the keychain by specifying a key identifier.

## Syntax

**key-id** *key-num*

**no key-id** *key-num*

## Command Default

A key is not configured by default.

## Parameters

*key-num*

Specifies the key identifier of the key. The valid range is from 1 through 4294967296.

## Modes

Keychain configuration mode

## Usage Guidelines

A maximum of 1024 keys can be configured across all the keychains.

Each key ID within a keychain has its own properties such as key string, authentication algorithm, send lifetime, and accept lifetime. A key is considered valid only if the key has a lifetime that has not expired, and the password and authentication algorithm have been specified.

The range of returned key IDs usable varies with the protocol. For each protocol, the key ID must be within a valid range. For example, the valid range of key IDs for OSPFv2 is from 1 through 255. The application that uses the keychain module can reject the key IDs that are outside the permitted range. However, the keychain module does not place any restrictions in terms of user configuration of the key ID.

The **no** form of the command deletes the key.

## Examples

The following example configures a key in a keychain.

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)# key-id 10
device(config-keychain-xprotocol-key-10)#
```

# History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# key-rollover-interval

Alters the timing of the existing configuration changeover.

## Syntax

**key-rollover-interval** *interval*

**no key-rollover-interval** *interval*

## Parameters

*interval*

Specifies the key-rollover-interval in seconds. Valid values range from 0 through 14400. The default is 300 seconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

In order to have consistent security parameters, rekeying should be done on all nodes at the same time. Use the **key-rollover-interval** command to facilitate this. The key rollover timer waits for a specified period of time before switching to the new set of keys. Use this command to ensure that all the nodes switch to the new set of keys at the same time.

The **no** form of the command resets the rollover interval to the default value of 300 seconds.

## Examples

The following example sets the key rollover interval to 420 seconds.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# key-rollover-interval 420
```

The following example re-sets the key rollover interval to the default value.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# no key-rollover-interval 420
```

The following example re-sets the key rollover interval to the default value in a nondefault VRF instance.

```
device# configure terminal
device(config)# ipv6 router ospf vrf red
device(config-ospf6-router-vrf-red))# no key-rollover-interval 420
```

# key-server-priority

Configures the MACsec key-server priority for the MACsec Key Agreement (MKA) group.

## Syntax

**key-server-priority** *value*

**no key-server-priority** *value*

## Command Default

Key-server priority is set to 16. This is not displayed in configuration details.

## Parameters

*value*

Specifies key-server priority. The possible values range from 0 to 255, where 0 is highest priority and 255 is lowest priority.

## Modes

dot1x-mka-cfg-group mode

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The **no** form of the command removes the previous priority setting.

During key-server election, the server with the highest priority (the server with the lowest key-server priority value) becomes the key-server.

## Examples

The following example sets the key-server priority for MKA group test1 to 5.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# key-server-priority 5
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.20a | This command was modified. The key-server priority value range was increased from 0 through 127 to 0 through 255. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |

| Release version | Command history |
|---|---|
| 08.0.70 | MACsec support was added on ICX 7650 devices. |

# kill

Terminates active CLI sessions.

## Syntax

**kill console** { **all** | *unit-number* }

**kill** { **ssh** | **telnet** } *session-number*

## Parameters

**console**
Logs out console sessions in a stack.

**all**
Logs out all console ports on stack units that are not the Active Controller.

*unit-number*
Logs out the console port on a specified unit.

**ssh**
Terminates an active SSH session.

**telnet**
Terminates an active Telnet session.

*session-number*
The Telnet or SSH session number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Once the AAA console is enabled, you should log out any open console ports on your traditional stack using the **kill console** command.

## Examples

The following example shows how to log out from all console ports on stack units that are not the Active Controller.

```
device# kill console all
```

The following example shows how to terminate an active SSH connection.

```
device# kill ssh 1
```

# lacp-timeout

Configures the timeout mode for the port.

## Syntax

**lacp-timeout** { **long** | **short** }

**no lacp-timeout** { **long** | **short** }

## Command Default

Begins with the short timeout period. Moves to the long timeout period after the LAG is established.

## Parameters

**long**

Specifies a long timeout period for the port which is 120 seconds.

**short**

Specifies a short timeout period for the port which is 3 seconds.

## Modes

LAG configuration mode

## Usage Guidelines

After you configure a port timeout mode, the port remains in that timeout mode whether it is up or down and whether or not it is part of a LAG. All the ports in a LAG must have the same timeout mode. This requirement is checked when the LAG is enabled on the ports.

With the long timeout configuration, an LACPDU is sent every 30 seconds. If no response comes from its partner after three LACPDUs are sent, a timeout event occurs, and the LACP state machine transitions to the appropriate state based on its current state.

In the short timeout configuration, an LACPDU is sent every second. If no response comes from its partner after three LACPDUs are sent, a timeout event occurs, and the LACP state machine transitions to the appropriate state based on its current state. If you do not include **long** or **short**, the device operates based on the IEEE specification standards.

**NOTE**
The configuration of lacp-timeout is applicable to dynamic or keep-alive LAGs only.

The **no** form of the command resets the timeout mode to short.

## Examples

The following example shows how to configure a port for a short LACP timeout.

```
device(config)# lag blue dynamic id 11
device(config-lag-blue)# lacp-timeout short
```

# lag

Creates a Link Aggregation Group (LAG).

## Syntax

**lag** *lag-name* { **dynamic** | **static** } { **id** { *number* | **auto** } }

**no lag** *lag-name* { **dynamic** | **static** } { **id** { *number* | **auto** } }

**lag** *lag-name* **keep-alive**

**no lag** *lag-name* **keep-alive**

## Command Default

LAG is not configured

## Parameters

*lag-name*
Specifies the name of the LAG as an ASCII string. The LAG name can be up to 64 characters in length.

**dynamic**
Configures a dynamic LAG.

**static**
Configures a static LAG.

**id** *number*
Specifies a LAG ID. The value ranges from 1 through 2047. The range is from 1 through 256 for user LAG and 256 and above for SPX LAG.

**auto**
Auto generates a LAG ID.

**keep-alive**
Configures a keep-alive LAG.

## Modes

Global configuration mode

## Usage Guidelines

The keep-alive LAG configuration can be used to configure a LAG for use in keep-alive applications similar to the UDLD.

A keep-alive LAG contains only one port while static and dynamic LAGs can have 1 to 8 or 1 to 12 ports depending on the device.

LAG IDs are unique for each LAG in the system. A LAG ID cannot be assigned to more than one LAG. If a LAG ID is already used, the CLI will reject the new LAG configuration and display an error message that suggests the next available LAG ID that can be used.

**NOTE**
The LAG ID parameter is applicable for static and dynamic LAGs only. No explicit configuration of a LAG ID is allowed on keep-alive LAGs.

The **no** form of the command removes the LAG.

# Examples

The following example shows how to configure a static LAG.

```
device(config)# lag blue static id 11
device(config-lag-blue)# ports ethernet 1/1/1 ethernet 1/1/5
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to make **id** option as a mandatory configuration parameter and also made the subsequent options (*id-number* and **auto**) mandatory. |

# lacp-mode passive

Configures Link Aggregation Control Protocol (LACP) operation mode for dynamic LAG as passive.

## Syntax

**lacp-mode passive**

**no lacp-mode passive**

## Command Default

The LACP operation mode is active.

## Parameters

**passive**

Configures LACP as passive. This option is applicable only for dynamic LAGs.

## Modes

LAG configuration mode

## Usage Guidelines

For dynamic LAGs, LACP is activated on all of the LAG ports.

The **no** form of the command changes the LACP operation mode of the dynamic LAG to the active mode (default mode).

## Examples

The following example configures LACP operation mode for dynamic LAG as passive.

```
device(config)# lag blue dynamic id 11
device(config-lag-blue)# lacp-mode passive
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# lag-mac

Assigns a static MAC address to the LAG virtual interface.

## Syntax

**lag-mac** *mac-address*

**no lag-mac** *mac-address*

## Command Default

The first physical port that is being added to the LAG becomes the MAC provider for the LAG virtual interface.

## Parameters

*mac-address*

Specifies the MAC address for the LAG virtual interface.

## Modes

LAG configuration mode

## Usage Guidelines

If VE/L3 is configured on the LAG, the **show interface brief** command output displays the stack-mac as the LAG virtual interface MAC.

The **no** form of the command removes the ststic MAC address assigned to the LAG virtual interface.

## Examples

The following example assigns a static MAC address to the LAG virtual interface.

```
device(config)# lag blue dynamic id 11
device(config-lag-blue)# lag-mac 0000.000f.e9a0
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# learn-default

Configures the device to learn default RIP routes, either globally or at the interface level.

## Syntax

**learn-default**

**ip rip learn-default**

**no learn-default** *device*/*slot*/*port*

**no ip rip learn-default** *device*/*slot*/*port*

## Command Default

By default, the device does not learn default RIP routes.

## Modes

RIP router configuration mode or interface configuration mode.

## Usage Guidelines

The **no** form of the command disables learning of default RIP routes.

The configurations at the global level and interface level are independent. Disabling or enabling one will not affect the other. When global level configuration is enabled, default routes are learned from all the interfaces. If global "learn-default " is not enabled but the interface-level "learn-default" is enabled, default routes are allowed from that rip interface. If "learn default" is not enabled for an interface, then the learned default routes for that interface are discarded.

# Examples

The following example enables learning of default RIP routes globally.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# learn-default
```

The following command output shows RIP default routes are learned globally.

```
device(config)# show ip rip
  RIP Summary
  Default port 520
      Administrative distance is 120
      Updates every 30 seconds, expire after 180
      Holddown lasts 180 seconds, garbage collect after 120
      Last broadcast  28, Next Update 30
      Need trigger update  0, Next trigger broadcast 4
      Minimum update  interval 25, Max update Offset 5
      Split horizon is on; poison reverse is off
      Import metric 1
      Default routes are accepted
      Prefix List, Inbound : Not set
      Prefix List, Outbound : Not set
      Route-map, Inbound : Not set
      Route-map, Outbound : Not set
      Redistribute:
  No Neighbors are configured in RIP Neighbor Filter Table
```

The following example enables learning of default RIP routes on Ethernet interface 1/1/6.

```
device# configure terminal
device(config)# interface ethernet 1/1/6
device(config-if-e10000-1/1/6)# ip rip learn-default
```

The following command output shows that RIP default routes are learned for the interface.

```
device(config)# show ip rip interface ethernet 1/1/16
Interface e 1/1/16
 RIP Mode : Version2 Running: TRUE
 Route summarization disabled
 Split horizon is on; poison reverse is off
 Default routes are accepted
 Metric-offset, Inbound 1
 Metric-offset, Outbound 0
 Prefix List, Inbound : Not set
      Prefix List, Outbound : Not set
 Route-map, Inbound : Not set
 Route-map, Outbound : Not set
 RIP Sent/Receive packet statistics:
    Sent : Request 0 Response 0
    Received : Total 0 Request 0 Response 0 UnRecognised 0
 RIP Error packet statistics:
    Rejected 0  Version 0 RespFormat 0 AddrFamily 0
    Metric 0 ReqFormat 0
```

# lease

Specifies the lease period for the DHCP address pool.

## Syntax

**lease** *days hours minutes*

## Parameters

*days hours minutes*

Specifies the lease duration in days, hours, and minutes.

## Modes

DHCP server pool configuration mode.

## Usage Guidelines

## Examples

The following example specifies the lease period as one day, four hours and 32 minutes.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# lease 1 4 32
```

# legacy-inline-power

Enables legacy power-consuming device detection globally, on multiple interfaces or on all ports of the stack or SPX.

## Syntax

**legacy-inline-power** [ **ethernet** *unit /slot/port* [ **to** *unit /slot/port* | [ **ethernet** *unit /slot/port* **to** *unit /slot/port* | **ethernet** *unit /slot/port* ]...] ]

**no legacy-inline-power** [ **ethernet** *unit /slot/port* [ **to** *unit /slot/port* | [ **ethernet** *unit /slot/port* **to** *unit /slot/port* | **ethernet** *unit /slot/port* ]...] ]

## Parameters

**ethernet** *unit/slot/port*

Enables legacy power-consuming device detection on specific interfaces.

**to** *unit/slot/port*

Specifies the range of ports on which you want to enable legacy power-consuming device detection.

## Command Default

PoE support for legacy power-consuming devices are not enabled by default.

## Modes

Global configuration mode

Stack configuration mode

## Usage Guidelines

Do not enable this command on ports where power-consuming devices are not connected.

With global configuration enabled, if the **legacy-inline-power** is configured at the interface level, it will be displayed in the interface level running configuration. Port-level legacy power-consuming device detection cannot be disabled from the global configuration mode. That is, when the **legacy-inline-power** configuration is removed globally (from enable configuration), it is not required for the user to configure **legacy-inline power** on the individual ports where it was already enabled.

By default, the **legacy-inline-power** command reserves 30 watts.

When the legacy support is disabled, 802.3af- and 802.3at-compliant devices are not affected.

The **no** form of the command disables support for PoE legacy power-consuming devices.

## Examples

The following example enables support for legacy power-consuming devices on multiple interfaces.

```
device(config)# legacy-inline-power ethernet 1/1/1 to 1/1/10
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was modified to allow legacy power-consuming device detection on multiple interfaces or to all ports of the stack or SPX. |

# legacy-inline-power (interface)

Enables legacy power-consuming device detection at the port level.

## Syntax

**legacy-inline-power**

**no legacy-inline-power**

## Command Default

PoE support for legacy power-consuming devices are not enabled by default.

## Modes

Interface subtype configuration mode

## Usage Guidelines

Do not enable this command on ports where power-consuming devices are not connected.

With global configuration enabled, if the **legacy-inline-power** is configured at the interface level, it will be displayed in the interface level running configuration. Port-level legacy power-consuming device detection cannot be disabled from the global configuration mode. That is, when the **legacy-inline-power** configuration is removed globally (from enable configuration), it is not required for the user to configure **legacy-inline power** on the individual ports where it was already enabled.

By default, the **legacy-inline-power** command reserves 30 watts.

When the legacy support is disabled, 802.3af- and 802.3at-compliant devices are not affected.

The **no** form of the command disables support for PoE legacy power-consuming devices.

## Examples

The following example enables support for legacy power-consuming devices on a specific interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# legacy-inline-power
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced at the interface configuration mode. |

# license delete perpetual

Deletes a software feature license and restores the default license option.

## Syntax

**license delete perpetual** *unit-ID license-name*

## Command Default

The feature license is installed.

## Parameters

*unit-id*

Specifies the unit ID on the device. For a standalone, the unit number is 1. For a stack unit, use the assigned stack unit number.

*license-name*

Specifies the license to be deleted. Options depend on the platform, but include the following:

- l3-prem
- macsec
- 2x10g
- 8x10g
- 2x10gr
- 4x10gr
- 8x10gr

## Modes

Privileged EXEC level

## Usage Guidelines

After deleting a license, the licensed feature will continue working until a reload.

## Examples

The following example shows a Layer 3 Premium license being deleted from an ICX 7250. Once a reload has been performed, the Layer 3 Premium features stop working and are no longer available, and the Layer 3 Base license is restored.

```
device# license delete perpetual 1 l3-prem
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# license delete unit

Deletes an XML software license and restores the default license option.

## Syntax

**license delete unit** *unit_id* [ **all** | **index** *license_index*]

## Parameters

*unit_id*

Specifies the unit ID number. For a standalone device, the unit number is 1. For a stack unit, use the assigned stack unit number.

**all**

Deletes all XML licenses on the specified unit.

**index** *license_index*

Specifies the software license file, and is generated by the member unit. The license index number is the license file you want to delete from a unit. The license index number is not unique across stack units, and you must specify both the unit number and the index number to delete a license from a specific unit. For example, an ICX7250-8X10G-LIC-POD license is installed on both stack unit 3, index 1, and stack unit 5, index 1. Because the index numbers are the same, you must specify both the unit number and the index number to delete a license from a specific unit.

## Modes

Privileged EXEC level.

## Usage Guidelines

Use the **license delete perpetual** command to delete an SAU license.

On the ICX 7450 and ICX 7250 devices, if more than one non-node locked license file is installed, the deletion of the license file sequence should start from the software license file identified by index 1. If this is not done, any attempt to delete the license file returns an error `141(LICENSE_IN_USE)`.

## Examples

The following example delete the license file as specified.

```
device# license delete unit 1 index 1
device# license delete unit 1 index 2
```

Use the all option to delete all license files for a specific unit.

```
device# license delete unit 1 all
```

# History

| Release version | Command history |
|---|---|
| 07.1.00 | This command was introduced. |
| 08.0.80 | The description was updated to indicate that it delete XML license files only. It does not delete SAU licenses. |

# license install perpetual

Installs a licensed feature on a device.

## Syntax

**license install perpetual** *unit-id license-package*

## Command Default

License install perpetual is disabled.

## Parameters

**unit-id**

Specifies the unit ID on the device. For a standalone, the unit number is 1. For a stack unit, use the assigned stack unit number.

*license-package*

Specifies the license name you have purchased and installed on the device. Options depend on the platform, but include the following:

- l3-prem
- macsec
- 2x10g
- 8x10g
- 2x10gr
- 4x10gr
- 8x10gr
- 4x1g
- 2x1g

## Modes

Privileged EXEC level

## Usage Guidelines

The command replaces the licensing commands for installing or deleting an XML license file.

## Examples

The following example installs a perpetual 4x10GR license on unit 1.

```
device# license install perpetual 1 4x10GR
```

The following example installs a perpetual 8X10GR license on an ICX 7150-48ZP serving as stack unit 3.

```
device# license install perpetual 3 8x10GR
```

# History

| Release version | Command history |
|---|---|
| 08.0.60 | This command was introduced. |
| 08.0.61 | Support for the command on an ICX 7150 stack was added. Support for an ICX 7150-48ZP license was added. |
| 08.0.80 | Support for this command was added to all the Ruckus ICX 7xxx series devices. |

# license set serial-number

Specifies the serial number of a software feature license on a device.

## Syntax

**license set serial-number** {*unit-ID license-type serial-number* }

## Command Default

The license serial number is not specified.

## Parameters

*unit-ID*

Specifies the unit ID on the device. For a standalone, the unit number is 1. For a stack unit, use the assigned stack unit number.

*license-type*

Specifies the type of SAU license. Options depend on the platform, but include the following:

- l3-prem
- pod
- macsec
- icx7150

*serial-number*

License serial number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Every license has a unique serial number. Once the license serial number has been specified on the device, it will appear in the **show license installed** command output and be available via SNMP.

The license serial number will be deleted if its corresponding license is deleted from the device.

## Examples

The following example sets the license serial number for a Layer 3 Premium license on unit 1 of the device:

```
device license set serial-number 1 l3-prem PR320400289
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# lifetime (IKEv2)

Configures the lifetime period of an Internet Key Exchange version 2 (IKEv2) security association (SA) for an IKEv2 profile.

## Syntax

**lifetime** *minutes*

**no lifetime** *minutes*

## Command Default

The default lifetime period for an IKEv2 SA is 43200 minutes (30 days).

## Parameters

*minutes*

Specifies the lifetime period (in minutes) for an IKEv2 SA. The range is from 10 through 43200.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

The **no** form of the command resets the IKEv2 SA lifetime period to the default value.

**NOTE**
During rekey, minor traffic loss is possible due to hardware programming delays.

## Examples

The following example shows how to set the IKEv2 SA lifetime value to 15000 minutes for an IKEv2 profile named prof-mktg.

```
device(config)# ikev2 profile prof-mktg
device(config-ike-profile-prof-mktg)# lifetime 15000
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# lifetime (IPsec)

Configures the lifetime period of an IPsec security association (SA) for an IPsec profile.

## Syntax

**lifetime** *minutes*

**no lifetime** *minutes*

## Command Default

The default lifetime period for an IPsec SA is 480 minutes (8 hours).

## Parameters

*minutes*

Specifies the lifetime period (in minutes) for an IPsec SA. The range is from 10 through 1440.

## Modes

IPsec profile configuration mode

## Usage Guidelines

Five minutes before an IPsec SA is due to expire, a new IPsec SA is started.

The **no** form of the command resets the IPsec SA lifetime period to the default value.

## Examples

The following example shows how to set the IPsec SA lifetime value to 720 minutes for an IPsec profile named prof-mktg.

```
device(config)# ipsec profile prof_mktg
device(config-ipsec-profile-prof_mktg)# lifetime 720
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# link-config gig copper autoneg-control

Configures the maximum advertised speed on a port that has auto-negotiation enabled.

## Syntax

**link-config gig copper autoneg-control** { **100m-auto** | **10m-auto** } { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no link-config gig copper autoneg-control** { **100m-auto** | **10m-auto** } { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The maximum port speed advertisement is not configured.

## Parameters

**100m-auto**
Configures a port to advertise a maximum speed of 100 Mbps.

**10m-auto**
Configures a port to advertise a maximum speed of 10 Mbps.

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]
Specifies the Ethernet interface, set of interfaces, or range of interfaces.

**to**
When followed by a port number, onfigures a range of ports.

## Modes

Global configuration mode

## Usage Guidelines

Maximum port speed advertisement is not supported on the Ruckus ICX 7750.

The maximum port speed advertisement works only when auto-negotiation is enabled (CLI command **speed-duplex auto**). If auto-negotiation is off, the device rejects the maximum port speed advertisement configuration.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The **no** form of the command disables the maximum port speed advertisement.

## Examples

The following command configures a maximum port speed advertisement of 10 Mbps on a port that has auto-negotiation enabled.

```
device(config)# link-config gig copper autoneg-control 10m-auto ethernet 1/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced for the Ruckus ICX 7450, but the **downshift** option was not supported. |
| 08.0.30 | This command was introduced for the Ruckus ICX 7250, but the **downshift** option was not supported. |
| 08.0.40 | The **downshift** option was removed. |

# link-error-disable

Configures port flap dampening on an interface.

## Syntax

**link-error-disable** *toggle-threshold sampling-time-in-sec wait-time-in-sec*

**no link-error-disable** *toggle-threshold sampling-time-in-sec wait-time-in-sec*

## Command Default

Port flap dampening is not configured.

## Parameters

*toggle-threshold*

Specifies the number of times a port link state goes from up to down and down to up before the wait period is activated. The value ranges from 1 through 50.

*sampling-time-in-sec*

Specifies the amount of time, in seconds, during which the specified toggle threshold can occur before the wait period is activated. The default value is 0 and indicates that the time is forever. The value ranges from 0 through 65535.

*wait-time-in-sec*

Specifies the amount of time, in seconds, for which the port remains disabled (down) before it becomes enabled. The value ranges from 0 through 65535. A value of 0 indicates that the port will stay down until an administrative override occurs.

## Modes

Interface configuration mode

## Usage Guidelines

A Ruckus device counts the number of times a port link state toggles from "up to down", and not from "down to up".

The sampling time or window (the time during which the specified toggle threshold can occur before the wait period is activated) is triggered when the first "up to down" transition occurs.

If the port link state toggles from up to down for a specified number of times within a specified period, the interface is physically disabled for the specified wait period. Once the wait period expires, the port link state is re-enabled. However, if the wait period is set to zero (0) seconds, the port link state will remain disabled until it is manually re-enabled.

You can configure the port flap dampening feature on the LAG virtual interface using the **link-error-disable** command. Once configured on the LAG virtual interface, the feature is enabled on all ports that are members of the LAG. You cannot configure port flap dampening on port members of the LAG.

The **no** form of the command re-enables a port that was disabled by port flap dampening once the wait period expires.

# Examples

The following example configures port flap dampening on an interface.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# link-error-disable 10 3 10
```

# link-fault-signal

Enables Link Fault Signaling (LFS) between 10 Gbps Ethernet devices.

## Syntax

**link-fault-signal**

**no link-fault-signal**

## Command Default

LFS is disabled by default on all Ruckus FastIron devices.

## Modes

Interface configuration mode

## Usage Guidelines

When configured on a Ruckus 10 Gbps Ethernet port, the port can detect and report fault conditions on transmit and receive ports. Ruckus recommends enabling LFS on both ends of a link.

Enable LFS on any device prior to connecting the device to FastIron platforms. Any connecting device must have LFS currently enabled to ensure interoperability. When LFS is enabled on an interface, syslog messages are generated when the link goes up or down, or when the TX or RX fiber is removed from one or both sides of the link that has LFS enabled.

You can view the status of an LFS-enabled link using the **show interface** command.

The **no** form of the command disables the Link Fault Signaling (LFS).

## Examples

The following example enables LFS.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# link-fault-signal
```

# link-keepalive ethernet

Enables UDLD for tagged and untagged control packets.

## Syntax

**link-keepalive ethernet** { *unit/slot/port* } [ **to** *unit/slot/port* [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **vlan** *vlan-ID* ]

**no link-keepalive ethernet** { *unit/slot/port* } [ **to** *unit/slot/port* [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **vlan** *vlan-ID* ]

## Command Default

UDLD is not enabled.

## Parameters

**ethernet** *unit/slot/port*
Specifies the Ethernet interface on which to enable UDLD.

**to** *unit/slot/port*
Specifies a range of Ethernet interfaces on which to enable UDLD.

**vlan** *vlan-ID*
Specifies the ID of the VLAN that the UDLD control packets can contain.

## Modes

Global configuration mode

## Usage Guidelines

UDLD is supported only on Ethernet ports.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

If you are specifying a VLAN ID, make sure that the VLAN ID is configured. A VLAN is specified when UDLD is configured. The port belongs to the configured VLAN as a tagged member. All the devices across the UDLD link are in the same VLAN. UDLD can be enabled on only one VLAN for a tagged port.

You must configure the same VLANs that will be used for UDLD on all devices across the network; otherwise, the UDLD link cannot be maintained.

You can specify a list of Ethernet ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The **no** form of the command disables UDLD for tagged and untagged control packets.

## Examples

The following example shows how to enable UDLD for untagged ports.

```
device(config)# link-keepalive ethernet 1/1/1
```

The following example shows how to configure UDLD on multiple ports.

```
device(config)# link-keepalive ethernet 1/1/1 ethernet 1/2/2
```

The following example shows how to configure UDLD on a range of ports.

```
device(config)# link-keepalive ethernet 1/1/1 to 1/1/5
```

The following example enables ports to receive and send UDLD control packets tagged with a specific VLAN ID.

```
device(config)# link-keepalive ethernet 1/1/8 vlan 22
```

# link-keepalive interval

Enables the interval time that UDLD sends health-check packets.

## Syntax

**link-keepalive interval** *time*

**no link-keepalive interval** *time*

## Command Default

By default, ports enabled for UDLD send a health-check packet once every 500 milliseconds (ms).

## Parameters

*time*

Specifies the time that UDLD sends the health-check packets, in milliseconds. You can specify from 1 through 60, in 100 ms increments (1 is 100 ms, 2 is 200 ms, and so on). The default is 5 (500 ms).

## Modes

Global configuration mode

## Usage Guidelines

A low UDLD link-keepalive interval is not recommended because low UDLD link-keepalive intervals are more sensitive and prone to flaps.

The **no** form of the command resets the interval to the default value.

## Examples

The following example shows the UDLD interval configuration.

```
device(config)# link-keepalive interval 4
```

# link-keepalive retries

Configures the maximum number of keep-alive attempts a port waits to receive a health-check reply packet from the port at the other end of the link.

## Syntax

**link-keepalive retries** *number*

**no link-keepalive retries** *number*

## Command Default

The default value is 7.

## Parameters

*number*

Specifies the number of keep-alive retries to receive a health-check reply packet. The valid range is from 3 through 64.

## Modes

Global configuration mode

## Usage Guidelines

By default, a port waits one second to receive a health-check reply packet from the port at the other end of the link. If the port does not receive a reply, the port tries six more times by sending up to six more health-check packets. If the port still does not receive a reply after the maximum number of retries, the port goes down.

The **no** form of the command changes the number of retries to the default value.

## Examples

The following example shows how to configure 10 retries as the maximum number of keep-alive attempts a port waits to receive a health-check reply packet.

```
device(config)# link-keepalive retries 10
```

# link-oam

Enables the EFM-OAM protocol and enters EFM-OAM protocol configuration mode.

## Syntax

**link-oam**

**no link-oam**

## Command Default

The EFM-OAM protocol is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

This command is not supported for ICX 7750 devices. The **no** form of the command removes all the EFM-OAM configurations.

## Examples

The following example enables EFM-OAM protocol configuration mode.

```
device# configure terminal
device(config)# link-oam
device(config-link-oam)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# lldp advertise link-aggregation

Advertises link-aggregation information.

## Syntax

**lldp advertise link-aggregation ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise link-aggregation ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] ... ] }

## Command Default

Link-aggregation information is automatically advertised when Link Layer Discovery Protocol (LLDP) is enabled on a global basis.

## Parameters

**ports**
: Advertises link-aggregation information for the specified ports.

**all**
: Advertises link-aggregation information for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises link-aggregation information for a specified Ethernet port.

**to** *unit/slot/port*
: Advertises link-aggregation information for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The devices advertise link-aggregation information about standard link aggregation (LACP) as well as static Link Aggregation (LAG) configuration.

The link-aggregation time, length, value (TLV) indicates the following:

- Whether the link is capable of being aggregated
- Whether the link is currently aggregated

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the link-aggregation advertisement.

# Examples

The following example enables advertisement of link-aggregation information for a specific Ethernet port.

```
device(config)# lldp advertise link-aggregation ports ethernet 1/1/1
```

# lldp advertise mac-phy-config-status

Advertises the MAC/PHY configuration and status.

## Syntax

**lldp advertise mac-phy-config-status ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise mac-phy-config-status ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The MAC/PHY configuration and status are automatically advertised when Link Layer Discovery Protocol (LLDP) is enabled on a global basis.

## Parameters

**ports**
: Advertises MAC/PHY configuration and status for the specified ports.

**all**
: Advertises MAC/PHY configuration and status for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises link-aggregation information for a specified Ethernet port.

**to** *unit/slot/port*
: Advertises link-aggregation information for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The MAC and PHY configuration and status of time, length, and value (TLV) includes the following information:

- Auto-negotiation capability and status.
- Speed and duplex mode.
- Flow control capabilities for auto-negotiation.
- Maximum port speed advertisement.
- If applicable, whether the above settings are the result of auto-negotiation during link initiation or a manual set override action.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the MAC/PHY advertisement.

# Examples

The following example enables the advertisement of MAC/PHY configuration and status for a specific Ethernet port.

```
device(config)# lldp advertise mac-phy-config-status ports ethernet 1/1/1
```

# lldp advertise management-address

Advertises a management address.

## Syntax

**lldp advertise management-address** { **ipv4** *ipv4-address* | **ipv6** *ipv6-address* } { **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] } }

**no lldp advertise management-address** { **ipv4** *ipv4-address* | **ipv6** *ipv6-address* } { **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] } }

## Command Default

Management address advertising has two modes: default and explicitly configured.

## Parameters

**ipv4** *ipv4-address*

Specifies an IPv4 management address to advertise.

**ipv6** *ipv6-address*

Specifies an IPv6 management address to advertise.

**ports**

Advertises the configured management address for the specified ports.

**all**

Advertises the configured management address for all LLDP-capable ports.

**ethernet** *unit/slot/port*

Advertises link-aggregation information for a specified Ethernet port.

**to** *unit/slot/port*

Advertises link-aggregation information for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The default mode is used when no addresses are configured to be advertised for a given port. If no management address is explicitly configured to be advertised, the device uses the first available IPv4 address and the first available IPv6 address (so it may advertise IPv4, IPv6 or both). If any addresses are configured to be advertised for a given port, then only those addresses are advertised. If no IP address is configured on any of the above, the port's current MAC address will be advertised.

If a management address is not explicitly configured to be advertised, the device uses the first available IPv4 address and the first available IPv6 address. A Layer 3 switch selects the first available address of each type from those configured on the following types of interfaces, in the following order of preference:

- Physical port on which LLDP will be transmitting the packet
- Virtual router interface (VE) on a VLAN that the port is a member of
- Dedicated management port
- Loopback interface
- Virtual router interface (VE) on any other VLAN
- Other physical port
- Other interface

For IPv6 addresses, link-local and anycast addresses are excluded from these searches.

If no IP address is configured on any of the listed interface types, the port's current MAC address is advertised.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command stops the advertisement of the management interface IP address.

# Examples

The following example advertises an IPv4 management address.

```
device(config)# lldp advertise management-address ipv4 10.157.2.1 ports ethernet 1/1/4
```

The following example advertises an IPv6 management address.

```
device(config)# lldp advertise management-address ipv6 2001:DB8::90 ports ethernet 1/1/7
```

# lldp advertise max-frame-size

Advertises the maximum frame size capability of the port.

## Syntax

**lldp advertise max-frame-size ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise max-frame-size ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The maximum frame size is automatically advertised when Link Layer Discovery Protocol (LLDP) is enabled on a global basis.

The maximum frame size is 1522 octets.

## Parameters

**ports**
Advertises the maximum frame size for the specified ports.

**all**
Advertises the maximum frame size for all LLDP-capable ports.

**ethernet** *unit/slot/port*
Advertises the maximum frame size for a specific Ethernet port.

**to** *stack-id/slot/port*
Advertises the maximum frame size for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The maximum frame size TLV provides the maximum 802.3 frame size capability of the port. This value is expressed in octets and includes the four-octet Frame Check Sequence (FCS). The default maximum frame size is 1522. The advertised value may change if the **aggregated-vlan** or **jumbo** command is configured.

**NOTE**
On 48GC modules in nonjumbo mode, the maximum size of ping packets is 1486 bytes and the maximum frame size of tagged traffic is no larger than 1581 bytes.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

# Examples

The following example enables the maximum frame size advertisement on a range of Ethernet ports.

```
device(config)# lldp advertise max-frame-size ports ethernet 1/1/4 to 1/1/12
```

# lldp advertise med-capabilities

Advertises information about Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED) capabilities.

## Syntax

**lldp advertise med-capabilities ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise med-capabilities ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

LLDP-MED information is automatically advertised when LLDP-MED is enabled.

## Parameters

**ports**
: Advertises LLDP-MED capabilities information for the specified ports.

**all**
: Advertises LLDP-MED capabilities information for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises LLDP-MED capabilities information for a specific Ethernet port.

**to** *unit/slot/port*
: Advertises LLDP-MED capabilities information for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The LLDP-MED capabilities advertisement includes the following information:

- The supported LLDP-MED TLVs
- The device type (network connectivity device or endpoint [Class 1, 2, or 3])

**NOTE**
Disabling the LLDP-MED capabilities disables LLDP-MED.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

# Examples

The following example enables the advertisement of LLDP-MED capabilities information on a range of Ethernet ports.

```
device(config)# lldp advertise med-capabilities ports ethernet 1/1/1 to 1/1/6
```

# lldp advertise med-power-via-mdi

Advertises endpoint IEEE 802.3af power-related information. Enables advanced power management between LLDP-MED endpoints and network connectivity devices.

## Syntax

**lldp advertise med-power-via-mdi ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise med-power-via-mdi ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

LLDP-MED power-via-MDI information is automatically advertised when LLDP-MED is enabled, when the port is a PoE port, and when PoE is enabled on the port.

## Parameters

**ports**
Advertises LLDP-MED power-via-MDI information for the specified ports.

**all**
Advertises LLDP-MED power-via-MDI information for all LLDP-capable ports.

**ethernet** *unit/slot/port*
Advertises LLDP-MED power-via-MDI information for a specific Ethernet interface.

**to** *unit/slot/port*
Advertises LLDP-MED power-via-MDI information for a range of Ethernet interfaces.

## Modes

Global configuration mode

## Usage Guidelines

The LLDP-MED Power-via-MDI TLV advertises an endpoint's IEEE 802.3af power-related information, including the following:

- Power type—whether the LLDP-MED device transmitting the LLPDU is a power-sourcing device or a powered device.
- Power source—The power source being utilized by a PSE or PD, for example, the primary power source, backup power source, or unknown.
- Power priority—The inline power priority level for the PSE or PD.
- Power level—The total power, in tenths of watts, required by a PD from a PSE or the total power that a PSE is capable of sourcing over a maximum length cable based on its current configuration.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

# Examples

The following example enables the advertisement of LLDP-MED power-via-MDI information for a range of Ethernet interfaces.

```
device(config)# lldp advertise med-power-via-mdi ports ethernet 1/1/1 to 1/1/5
```

# lldp advertise port-description

Identifies the port from which the Link Layer Discovery Protocol (LLDP) agent transmitted the advertisement.

## Syntax

**lldp advertise port-description ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise port-description ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The port description is automatically advertised when LLDP is enabled on a global basis.

## Parameters

**ports**
: Advertises the port description for the specified ports.

**all**
: Advertises the port description for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises the port description for a specific Ethernet port.

**to** *unit/slot/port*
: Advertises the port description for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The port description is taken from the ifDescr MIB object from MIB-II.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

## Examples

The following example enables the advertisement of the port description on a range of Ethernet ports.

```
device(config)# lldp advertise port-description ports ethernet 1/1/4 to 1/1/9
```

# lldp advertise port-id-subtype

Specifies the Link Layer Discovery Protocol (LLDP) port ID subtype information to advertise as the port ID.

## Syntax

**lldp advertise port-id-subtype** *num* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise port-id-subtype** *num* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

By default, the port ID subtype to advertise is set to 3.

## Parameters

*num*

Specifies the port ID subtype to advertise. The subtype determines the specific information that is advertised as the port ID.

**1**

Causes interface alias information, taken from the ifAlias MIB object, to be advertised as the port ID.

**3**

Causes the MAC address to be advertised as the port ID. This is the default value.

**5**

Causes interface name information, taken from the ifName MIB object, to be advertised as the port ID.

**7**

Causes a locally assigned value (as defined by RFC 2863) to be displayed as the port ID. Ruckus devices display information taken from the ifIndex MIB object.

**ports**

Specifies the LLDP-capable ports for which the LLDP port ID subtype is to be advertised.

**all**

Causes the advertisement of the port ID subtype for all LLDP-capable ports on the device.

**ethernet** *unitslot/port*

Causes the advertisement of the port ID subtype for a specific Ethernet port. When immediatly followed by the **to** option, this option specifies the first port in a range of Ethernet ports.

**to** *unit/slot/port*

Causes the advertisement of the port ID subtype for a range of Ethernet ports and specifies the last port in the range.

**NOTE**

You can specify the advertisement of an LLDP port ID subtype for a range of Ethernet ports (for example, ethernet 1/1/1 to ethernet 1/1/4), or for a list of Ethernet ports (for example, ethernet 1/2/1 ethernet 1/2/2), or you can combine a range with a list (for example, ethernet 1/1/1 to ethernet 1/1/4 ethernet 1/1/1 ethernet 1/1/2).

# Modes

Global configuration mode

# Usage Guidelines

**NOTE**
The port ID subtype to advertise is only configurable on Ruckus ICX 7750, Ruckus ICX 7450, and Ruckus ICX 7250 devices.

The LLDP port ID subtype advertises previously configured information. To ensure that the physical location of a port is available for advertisement when the port ID subtype to advertise is set to 1, 5, or 7, the port location must first be configured by using the **lldp med location-id civic-address**, **lldp med location-id coordinate-based**, or **lldp med location-id ecs-elin** command.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command restores the port ID subtype advertised to the default value for specific ports.

# Examples

The following example shows how to advertise the interface alias (port ID subtype 1) as the port ID for two individual ports (1/2/1 and 1/2/2) and for a range of ports (1/1/1 to 1/1/4).

```
device(config)# lldp advertise port-id-subtype 1 ports ethernet 1/2/1 ethernet 1/2/2 ethernet 1/1/1 to
1/1/4
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# lldp advertise port-vlan-id

Advertises the port VLAN identifier (PVID) that is associated with untagged or priority-tagged frames.

## Syntax

**lldp advertise port-vlan-id ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise port-vlan-id ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The port VLAN ID is automatically advertised when Link Layer Discovery Protocol (LLDP) is enabled on a global basis.

## Parameters

**ports**
: Advertises the port VLAN ID for the specified ports.

**all**
: Advertises the port VLAN ID for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises the port VLAN ID for a specific Ethernet port.

**to** *unit/slot/port*
: Advertises the port VLAN ID for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

## Examples

The following example enables the advertisement of the port VLAN ID on a range of ports.

```
device(config)# lldp advertise port-vlan-id ports ethernet 1/1/2 to 1/1/5
```

# lldp advertise power-via-mdi

Advertises general information about Power over Ethernet (PoE) capabilities and the status of the port.

## Syntax

**lldp advertise power-via-mdi ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise power-via-mdi ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

Information about PoE capabilities and port status is not advertised.

## Parameters

**ports**
: Advertises Power via Media Dependent Interface (power-via-MDI) information for the specified ports.

**all**
: Advertises power-via-MDI information for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises power-via-MDI information for a specific Ethernet port.

**to** *unit/slot/port*
: Advertises power-via-MDI information for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The power-via-MDI information includes the following:

- PoE capability (supported or not supported)
- PoE status (enabled or disabled)
- Power Sourcing Equipment (PSE) power pair—Indicates which pair of wires is in use and whether the pair selection can be controlled. The Ruckus implementation always uses pair A and cannot be controlled.
- Power class—Indicates the range of power that the connected powered device has negotiated or requested.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

# Examples

The following example advertises the power-via-MDI information on a range of ports.

```
device(config)# lldp advertise power-via-mdi ports ethernet 1/1/1 to 1/1/10
```

# lldp advertise system-capabilities

Advertises the primary functions of the device and indicates whether these primary functions are enabled.

## Syntax

**lldp advertise system-capabilities ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise system-capabilities ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The system capabilities are automatically advertised when Link Layer Discovery Protocol (LLDP) is enabled on a global basis.

## Parameters

**ports**
: Advertises the system capabilities for the specified ports.

**all**
: Advertises the system capabilities for all LLDP-capable ports.

**ethernet** *unitslot/port*
: Advertises the system capabilities for the specified Ethernet port.

**to** *unit/slot/port*
: Advertises the system capabilities for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

System capabilities are based on the type of software image in use (Layer 2 switch or Layer 3 router). The enabled capabilities are the same as the available capabilities, except that when using a router image (base or full Layer 3), if the global route-only feature is turned on, the bridge capability is not included, since no bridging occurs.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

## Examples

The following example advertises the system capabilities information on a range of ports.

```
device(config)# lldp advertise system-capabilities ports ethernet 1/1/1 to 1/1/10
```

# lldp advertise system-description

Advertises information such as the product name or model number, the version of the system hardware, the software operating system level, and the networking software version.

## Syntax

**lldp advertise system-description ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise system-description ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The system description is not advertised.

## Parameters

**ports**
: Advertises the system information for the specified ports.

**all**
: Advertises the system information for all Link Layer Discovery Protocol (LLDP) capable ports.

**ethernet** *unitslot/port*
: Advertises the system information for a specific Ethernet port.

**to** *unit/slot/port*
: Advertises the system information for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The system description is the network entity, which can include information such as the product name or model number, the version of the system hardware, the software operating system level, and the networking software version. The information corresponds to the sysDescr MIB object in MIB-II.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

## Examples

The following example advertises the system description information.

```
device(config)# lldp advertise system-description ports ethernet 1/1/1 to 1/1/5
```

# lldp advertise system-name

Advertises the name assigned to the system.

## Syntax

**lldp advertise system-name ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp advertise system-name ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

The system name is automatically advertised when Link Layer Discovery Protocol (LLDP) is enabled on a global basis.

## Parameters

**ports**
: Advertises the system name for the specified ports.

**all**
: Advertises the system name for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Advertises the system name for a specific Ethernet port.

**to** *unit/slot/port*
: Advertises the system name for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

The system name is the name that is administratively assigned to the system and is taken from the sysName MIB object in MIB-II. The sysName MIB object corresponds to the name defined with the CLI command **hostname**.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the advertisement.

## Examples

The following example advertises the system name information.

```
device(config)# lldp advertise system-name ports ethernet 1/1/1 to 1/1/0
```

# lldp enable ports

Enables the receipt and transmission of Link Layer Discovery Protocol (LLDP) packets on ports.

## Syntax

**lldp enable ports ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp enable ports ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

When LLDP is enabled on a global basis, by default, each port on a Ruckus device is capable of transmitting and receiving LLDP packets.

## Parameters

**ports**
: Enables LLDP for the specified ports.

**all**
: Enables LLDP for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Enables LLDP for a specifc Ethernet port.

**to** *unit/slot/port*
: Enables LLDP for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

When a port is configured to both receive and transmit LLDP packets and the MED capabilities TLV is enabled, LLDP-MED is enabled as well. LLDP-MED is not enabled if the operating mode is set to receive only or transmit only.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables the receipt and transmission of LLDP packets on the specified ports.

## Examples

The following example enables LLDP on one port.

```
device(config)# lldp enable ports ethernet 1/1/1
```

# lldp enable receive

Changes the Link Layer Discovery Protocol (LLDP) operating mode of specified ports from receive-and-transmit mode to receive-only mode.

## Syntax

**lldp enable receive ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp enable receive ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] ... ] }

## Command Default

When LLDP is enabled on a global basis, each port on the device is capable of transmitting and receiving LLDP packets.

## Parameters

**ports**
Changes the LLDP operating mode to receive-only mode for the specified ports.

**all**
Changes the LLDP operating mode to receive-only mode for all LLDP-capable ports.

**ethernet** *unit/slot/port*
Changes the LLDP operating mode to receive-only mode for a specific Ethernet port.

**to** *unit/slot/port*
Changes the LLDP operating mode to receive-only mode for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

To change the LLDP operating mode to transmit-only mode, disable the receive mode using the **no lldp enable transmit** command.

**NOTE**
LLDP-MED is not enabled when you enable the receive-only operating mode. To enable LLDP-MED, you must configure the port to both receive and transmit LLDP packets.

**NOTE**
To change a port's LLDP operating mode from transmit-only to receive-only, first disable the transmit-only mode, and then enable the receive-only mode. If you do not disable transmit-only mode, you will configure the port to both receive and transmit LLDP packets.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command changes the LLDP operating mode to transmit-only mode if the device is in both transmit and receive mode, and it disables the LLDP receive-only operating mode if receive-only mode was enabled.

# Examples

The following example changes the LLDP operating mode of three ports to receive-only mode.

```
device(config)# lldp enable receive ports ethernet 1/1/1 ethernet 1/1/5 ethernet 1/1/7
```

# lldp enable snmp med-topo-change-notifications

Enables SNMP notifications and syslog messages for LLDP-MED topology changes.

## Syntax

**lldp enable snmp med-topo-change-notifications ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp enable snmp med-topo-change-notifications ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

SNMP notifications and corresponding syslog messages are disabled.

## Parameters

**ports**
Enables LLDP-MED SNMP notifications and syslog messages for ports.

**all**
Enables LLDP-MED SNMP notifications and syslog messages for all LLDP-capable ports.

**ethernet** *unit/slot/port*
Enables LLDP-MED SNMP notifications and syslog messages for a specific Ethernet port.

**to** *unit/slot/port*
Enables LLDP-MED SNMP notifications and syslog messages for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

When you enable LLDP-MED SNMP notifications, corresponding syslog messages are enabled as well. When you enable LLDP-MED SNMP notifications, the device sends traps and syslog messages when an LLDP-MED endpoint neighbor entry is added or removed.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables LLDP-MED SNMP notifications and syslog messages.

## Examples

The following example enables LLDP-MED SNMP notifications and syslog messages for a range of ports.

```
device(config)# lldp enable snmp med-topo-change-notifications ports ethernet 1/1/4 to 1/1/6
```

# lldp enable snmp notifications

Enables LLDP SNMP notifications and syslog messages.

## Syntax

**lldp enable snmp notifications ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp enable snmp notifications ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

LLDP SNMP notifications and corresponding syslog messages are disabled.

## Parameters

**ports**
: Enables LLDP SNMP notifications and syslog messages for ports.

**all**
: Enables LLDP SNMP notifications and syslog messages for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Enables LLDP SNMP notifications and syslog messages for a specific Ethernet port.

**to** *unit/slot/port*
: Enables LLDP SNMP notifications and syslog messages for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

When you enable LLDP SNMP notifications, the device sends traps and corresponding syslog messages whenever there is a change in the LLDP data received from neighboring devices.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command disables LLDP SNMP notifications and syslog messages.

## Examples

The following example enables LLDP SNMP notifications and syslog messages for a range of ports.

```
device(config)# lldp enable snmp notifications ports ethernet 1/1/1 to 1/1/6
```

# lldp enable transmit

Changes the Link Layer Discovery Protocol (LLDP) operating mode to transmit-only mode.

## Syntax

**lldp enable transmit ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp enable transmit ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

When LLDP is enabled on a global basis, each port on the device is capable of transmitting and receiving LLDP packets.

## Parameters

**ports**
Changes the LLDP operating mode to transmit-only mode for ports.

**all**
Changes the LLDP operating mode to transmit-only mode for all LLDP-capable ports.

**ethernet** *unit/slot/port*
Changes the LLDP operating mode to transmit-only mode for the specified Ethernet interface.

**to** *unit/slot/port*
Changes the LLDP operating mode to transmit-only mode for a range of Ethernet interfaces.

## Modes

Global configuration mode

## Usage Guidelines

**NOTE**
To change a port's LLDP operating mode from receive- only to transmit-only, first disable receive-only mode, and then enable transmit-only mode. If you do not disable receive-only mode, you will configure the port to both receive and transmit LLDP packets.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command changes the LLDP operating mode to receive-only mode if the device is in both transmit and receive mode, and it disables the LLDP transmit-only operating mode if transmit-only mode was enabled.

# Examples

The following example sets the LLDP operating mode to transmit-only mode.

```
device(config)# no lldp enable receive ports ethernet 1/1/1 ethernet 1/1/8
device(config)# lldp enable transmit ports ethernet 1/1/1 ethernet 1/1/8
```

# lldp max-neighbors-per-port

Specifies the maximum number of Link Layer Discovery Protocol (LLDP) neighbors per port.

## Syntax

**lldp max-neighbors-per-port** *value*

**no lldp max-neighbors-per-port**

## Command Default

The default number of LLDP neighbors per port is 4.

## Parameters

*value*

Specifies the number of LLDP neighbors for which LLDP data is retained for each port. The value can range from 1 through 64. The default value is 4.

## Modes

Global configuration mode

## Usage Guidelines

You can use the **show lldp** command to view the current configuration.

The **no** form of the command removes the configured value and restores the default value of 4.

## Examples

The following example sets the number of LLDP neighbors per port to 6.

```
device(config)# lldp max-neighbors-per-port 6
```

# lldp max-total-neighbors

Specifies the maximum number of Link Layer Discovery Protocol (LLDP) neighbors for which LLDP data is retained for the entire system.

## Syntax

**lldp max-total-neighbors** *value*

**no lldp max-total-neighbors**

## Command Default

The default number of LLDP neighbors per device is 392.

## Parameters

*value*

Specifies the number of LLDP neighbors per device. The value can range from 16 through 8192. The default value is 392.

## Modes

Global configuration mode

## Usage Guidelines

You can use the **show lldp** command to view the current configuration.

The **no** form of the command removes the configured value and restores the default value of 392 LLDP neighbors.

## Examples

The following example sets the number of LLDP neighbors per device to 100.

```
device(config)# lldp max-total-neighbors 100
```

# lldp med fast-start-repeat-count

Configures the Link Layer Discovery Protocol Media Endpoint Device (LLDP-MED) fast-start transmit count.

## Syntax

**lldp med fast-start-repeat-count** *value*

**no lldp med fast-start-repeat-count**

## Command Default

The device sends three packets at 1-second intervals.

## Parameters

*value*

Specifies the number of LLDP packets that are sent during the LLDP-MED fast-start period. The value can range from 1 through 10. The default value is 3.

## Modes

Global configuration mode

## Usage Guidelines

The LLDP-MED fast-start repeat count specifies the number of LLDP packets that are sent during the LLDP-MED fast-start period.

The fast-start feature enables a network connectivity device to initially advertise itself at a faster rate for a limited time when an LLDP-MED endpoint has been newly detected or has newly connected to the network. This feature is important within a VoIP network, for example, where rapid availability is crucial for applications such as emergency call service location (E911). The fast-start timer starts when a network connectivity device receives the first LLDP frame from a newly detected endpoint.

**NOTE**
The LLDP-MED fast-start mechanism is intended to run only on links between network connectivity devices and endpoint devices. It does not apply to links between LAN infrastructure elements, including between network connectivity devices or to other types of links.

The **no** form of the command removes the configured value and restores the default value of 3 packets per second.

## Examples

The following example sets the LLDP-MED fast-start transmit count to 6.

```
device(config)# lldp med fast-start-repeat-count 6
```

# lldp med location-id civic-address

Configures a civic-address-based location for Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED).

## Syntax

**lldp med location-id civic-address refers-to** *reference* **country** *country-code* [ **elem** *CA-type value* [ **elem** *CA-type value* ] ... ] **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp med location-id civic-address refers-to** *reference* **country** *country-code* [ **elem** *CA-type value* [ **elem** *CA-type value* ] ... ] **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

An LLDP-MED civic address is not configured.

## Parameters

**refers-to** *reference*

Specifies the location that the entry refers to. Specify one of the following: **client**, **dhcp-server**, or **network-element**.

**NOTE**
The **dhcp-server** or **network-element** keywords should be used only if it is known that the endpoint is in close physical proximity to the DHCP server or network element.

**country** *country-code*

Specifies a two-letter ISO 3166 country code in capital ASCII letters as follows:

- **CA** (Canada)
- **DE** (Germany)
- **JP** (Japan)
- **KR** (Korea)
- **US** (United States)

**elem** *CA-type*

Specifies the civic address element. This a value from 0 to 255 that describes the civic address element. Refer to the usage guidelines.

*value*

Specifies the actual value of the element CA type.

**ethernet** *unit/slot/port*

Specifies the Ethernet port.

**to** *unit/slot/port*

Specifies a range of Ethernet ports.

## Modes

Global configuration mode

# Usage Guidelines

If the value of an element contains one or more spaces, use double quotation marks (") at the beginning and end of the string. For example, **elem 3 "Santa Clara"**.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command removes the configuration.

**TABLE 7 Elements used with a civic address**

| Civic address (CA) type | Description | Acceptable values / examples |
|---|---|---|
| 0 | Language | The ISO 639 language code used for presenting the address information. |
| 1 | National subdivisions (state, canton, region, province, or prefecture) | Examples:<br>Canada - Province<br>Germany - State<br>Japan - Metropolis<br>Korea - Province<br>United States - State |
| 2 | County, parish, gun (JP), or district (IN) | Examples:<br>Canada - County<br>Germany - County<br>Japan - City or rural area<br>Korea - County<br>United States - County |
| 3 | City, township, or shi (JP) | Examples:<br>Canada - City or town<br>Germany - City<br>Japan - Ward or village<br>Korea - City or village<br>United States - City or town |
| 4 | City division, borough, city district, ward, or chou (JP) | Examples:<br>Canada - N/A<br>Germany - District<br>Japan - Town<br>Korea - Urban district<br>United States - N/A |
| 5 | Neighborhood or block | Examples:<br>Canada - N/A<br>Germany - N/A<br>Japan - City district<br>Korea - Neighborhood |

**TABLE 7 Elements used with a civic address (continued)**

| Civic address (CA) type | Description | Acceptable values / examples |
|---|---|---|
| | | United States - N/A |
| 6 | Street | Examples:<br>Canada - Street<br>Germany - Street<br>Japan - Block<br>Korea - Street<br>United States - Street |
| 16 | Leading street direction | N (north), E (east), S (south), W (west), NE, NW, SE, SW |
| 17 | Trailing street suffix | N (north), E (east), S (south), W (west), NE, NW, SE, SW |
| 18 | Street suffix | Acceptable values for the United States are listed in the United States Postal Service Publication 28 [18], Appendix C.<br>Example: Ave, Place |
| 19 | House number | The house number (street address)<br>Example: 1234 |
| 20 | House number suffix | A modifier to the house number. It does not include parts of the house number.<br>Example: A, 1/2 |
| 21 | Landmark or vanity address | A string name for a location. It conveys a common local designation of a structure, a group of buildings, or a place that helps to locate the place.<br>Example: UC Berkeley |
| 22 | Additional location information | An unstructured string name that conveys additional information about the location.<br>Example: west wing |
| 23 | Name (residence and office occupant) | Identifies the person or organization associated with the address.<br>Example: Textures Beauty Salon |
| 24 | Postal / zip code | The valid postal / zip code for the address.<br>Example: 95054-1234 |
| 25 | Building (structure) | The name of a single building if the street address includes more than one building or if the building name is helpful in identifying the location.<br>Example: Law Library |
| 26 | Unit (apartment, suite) | The name or number of a part of a structure where there are separate administrative units, owners, or tenants, such as separate companies or families who occupy that structure. Common examples include suite or apartment designations.<br>Example: Apt 27 |
| 27 | Floor | Example: 4 |
| 28 | Room number | The smallest identifiable subdivision of a structure.<br>Example: 7A |

**TABLE 7 Elements used with a civic address (continued)**

| Civic address (CA) type | Description | Acceptable values / examples |
|---|---|---|
| 29 | Place type | The type of place described by the civic coordinates. For example, a home, office, street, or other public space.<br>Example: Office |
| 30 | Postal community name | When the postal community name is defined, the civic community name (typically CA type 3) is replaced by this value.<br>Example: Alviso |
| 31 | Post office box (P.O. box) | When a P.O. box is defined, the street address components (CA types 6, 16, 17, 18, 19, and 20) are replaced with this value.<br>Example: P.O. Box 1234 |
| 32 | Additional code | An additional country-specific code that identifies the location. For example, for Japan, this is the Japan Industry Standard (JIS) address code. The JIS address code provides a unique address inside of Japan, down to the level of indicating the floor of the building. |
| 128 | Script | The script (from ISO 15924 [14]) used to present the address information.<br>Example: Latn<br>**NOTE**<br>If not manually configured, the system assigns the default value **Latn**. |
| 255 | Reserved | |

The **no** form of the command removes the LLDP-MED civic address.

# Examples

The following example configures a civic-address-based location.

```
device(config)# lldp med location-id civic-address refers-to client country US elem 1 CA elem 3 "Santa
Clara" elem 6 "4980 Great America Pkwy" elem 24 95054 elem 27 5 elem 28 551 elem 29 office elem 23
"John Doe"
```

# lldp med location-id coordinate-based

Configures a coordinate-based location for an endpoint device.

## Syntax

**lldp med location-id coordinate-based latitude** *degrees* **resolution** *bits* **longitude** *degrees* **resolution** *bits* **altitude** { **floors** *number* **resolution** *bits* | **meters** *number* **resolution** *bits* } *datum* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp med location-id coordinate-based latitude** *degrees* **resolution** *bits* **longitude** *degrees* **resolution** *bits* **altitude** { **floors** *number* **resolution** *bits* | **meters** *number* **resolution** *bits* } *datum* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

A coordinate-based location for an endpoint device is not configured.

## Parameters

**latitude** *degrees*

Specifies the angular distance north or south from the earth equator, measured through 90 degrees. Positive numbers indicate a location north of the equator and negative numbers indicate a location south of the equator.

**resolution** *bits*

Specifies the precision of the value given for latitude. A smaller value increases the area within which the device is located. For latitude, the value can range from 1 to 34.

**longitude** *degrees*

Specifies the angular distance from the intersection of the zero meridian. Positive values indicate a location east of the prime meridian and negative numbers indicate a location west of the prime meridian.

**resolution** *bits*

Specifies the precision of the value given for longitude. A smaller value increases the area within which the device is located. For longitude resolution, enter a number between 1 and 34.

**altitude**

Specifies the vertical elevation of a building above the ground.

**floors** *number*

Specifies the vertical elevation of a building above the ground, where 0 represents the floor level associated with the ground level at the main entrance and larger values represent floors that are above (higher in altitude) floors with lower values. Subfloors can be represented by noninteger values.

**resolution** *bits*

Specifies the precision of the value given for altitude. A smaller value increases the area within which the device is located. For floor resolution, enter the value 0 if the floor is unknown or 30 if a valid floor is being specified.

**meters** *number*

Specifies the vertical elevation, in meters, as opposed to floors.

**resolution** *bits*
Specifies the precision of the value given for altitude. A smaller value increases the area within which the device is located. For meter resolution, enter a value from 0 to 30.

*datum*
Specifies the map used as the basis for calculating the location. The value can be one of the following:

**wgs84**
World Geodesic System 1984, CRS Code 4327, Prime Meridian Name: Greenwich.

**nad83-navd88**
North American Datum 1983, CRS Code 4269, Prime Meridian Name: Greenwich. The associated vertical datum is the North American Vertical Datum of 1988 (NAVD88). Use this value when referencing locations on land. If land is near tidal water, use **nad83-mllw**.

**nad83-mllw**
North American Datum 1983, CRS Code 4269, Prime Meridian Name: Greenwich. The associated vertical datum is mean lower low water (MLLW). Use this value when referencing locations on water, sea, or ocean.

**ports**
Introduces the set of Ethernet interfaces to be included in the configuration.

**all**
Specifies that all Ethernet ports included in the configuration.

**ethernet** *unit/slot/port*
Specifies an Ethernet port.

**to** *unit/slot/port*
Specifies a range of Ethernet ports.

# Modes

Global configuration mode

# Usage Guidelines

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command removes a coordinate-based location for an Endpoint device.

# Examples

The following example configures a coordinate-based location.

```
device(config)# lldp med location-id coordinate-based latitude -78.303 resolution 20 longitude 34.27
resolution 18 altitude meters 50 resolution 16 wgs84 ports all
```

# lldp med location-id ecs-elin

Configures an Emergency Call Service (ECS) based location for Link Layer Discovery Protocol Media Endpoint Device (LLDP-MED).

## Syntax

**lldp med location-id ecs-elin** *numeric-string* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp med location-id ecs-elin** *numeric-string* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Parameters

*numeric-string*
: Specifies the Emergency Location Identification Number (ELIN) from the North America Numbering Plan format, supplied to the Public Safety Answering Point (PSAP) for ECS purposes. The value can range from 10 to 25 digits in length.

**ports**
: Configures an ECS-based location for ports.

**all**
: Configures an ECS-based location for all LLDP-capable ports.

**ethernet** *unit/slot/port*
: Configures an ECS-based location for a specific Ethernet port.

**to** *unit/slot/port*
: Configures an ECS-based location for a range of Ethernet ports.

## Modes

Global configuration mode

## Usage Guidelines

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command removes the configured ECS-based location.

## Examples

The following example configures an ECS-based location for LLDP-MED.

```
device(config)# lldp med location-id ecs-elin 4082071700 ports ethernet 1/2/1 to 1/2/4
```

# lldp med network-policy application

Defines an Link Layer Discovery Protocol-Media Endpoint Discovery (LLDP-MED) network policy for an endpoint.

## Syntax

**lldp med network-policy application** *application-type* **tagged vlan** *vlan-id* **priority** *priority-value* **dscp** *dscp-value* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp med network-policy application** *application-type* **tagged vlan** *vlan-id* **priority** *priority-value* **dscp** *dscp-value* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**lldp med network-policy application** *application-type* **untagged dscp** *dscp-value* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp med network-policy application** *application-type* **untagged dscp** *dscp-value* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**lldp med network-policy application** *application-type* **priority-tagged priority** *priority-value* **dscp** *dscp-value* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

**no lldp med network-policy application** *application-type* **priority-tagged priority** *priority-value* **dscp** *dscp-value* **ports** { **all** | [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] }

## Command Default

An LLDP-MED network policy is not defined.

## Parameters

*application-type*

Configures the primary function of the applications defined by this network policy. The application type can be one of the following:

**guest-voice**

Limited voice service for guest users and visitors with their own IP telephony handsets or similar devices that support interactive voice services.

**guest-voice-signaling**

Limited voice service for use in network topologies that require a different policy for guest voice signaling than for guest voice media.

**softphone-voice**

Softphone voice service for use with multimedia applications that work in association with VoIP technology, enabling phone calls direct from a PC or laptop. Softphones do not usually support multiple VLANs and are typically configured to use an untagged VLAN or a single tagged data-specific VLAN. Note that when a network policy is defined for use with an untagged VLAN, the Layer 2 priority field is ignored and only the DSCP value is relevant.

**streaming-video**

Applies to broadcast- or multicast-based video content distribution and similar applications that support streaming video services requiring specific network policy treatment. Video applications that rely on TCP without buffering would not be an intended use of this application type.

**video-conferencing**
Applies to dedicated video conferencing equipment and similar devices that support real-time interactive video/audio services.

**video-signaling**
For use in network topologies that require a separate policy for video signaling than for video media. Note that this application type should not be advertised if all the same network policies apply as those advertised in the video conferencing policy TLV.

**voice**
For use by dedicated IP telephony handsets and similar devices that support interactive voice services.

**voice-signaling**
For use in network topologies that require a different policy for voice signaling than for voice media. Note that this application type should not be advertised if all the same network policies apply as those advertised in the voice policy TLV.

**tagged vlan** *vlan-id*
Specifies the tagged VLAN that the specified application type will use.

**untagged**
Configures the device to use an untagged frame format.

**priority-tagged**
Configures the device to use priority-tagged frames. In this case, the device uses the default VLAN (PVID) of the ingress port.

**priority** *priority-value*
Configures the Layer 2 priority value to be used for the specified application type. Enter 0 to use the default priority. Valid values are 0 through 7.

**dscp** *dscp-value*
Configures the Layer 3 differentiated services codepoint priority value to be used for the specified application type. Enter 0 to use the default priority. Valid values are 0 through 63.

**ports**
Specifies the ports.

**ethernet** *unit/slot/port*
Configures the network policy on the specified Ethernet interface.

**to** *unit/slot/port*
Configures the network policy on a range of Ethernet interfaces.

# Modes

Global configuration mode

# Usage Guidelines

An LLDP-MED network policy defines an endpoint VLAN configuration (VLAN type and VLAN ID) and associated Layer 2 and Layer 3 priorities that apply to a specific set of applications on a port.

**NOTE**
This feature applies to applications that have specific real-time network policy requirements, such as interactive voice or video services. It is not intended to run on links other than links between network connectivity devices and endpoints, and therefore does not advertise the multitude of network policies that frequently run on an aggregated link.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

The command cannot support LAG virtual interfaces. Attempting to configure them will have no effect.

The **no** form of the command removes the defined LLDP-MED network policy for an Endpoint.

# Examples

The following example defines an LLDP-MED network policy for an endpoint.

```
device(config)# lldp med network-policy application voice tagged vlan 99 priority 3 dscp 22 ports
ethernet 1/1/1 to 1/1/3
```

# lldp reinit-delay

Configures the minimum time between port reinitializations.

## Syntax

**lldp reinit-delay** *seconds*

**no lldp reinit-delay**

## Command Default

When LLDP is enabled, the default time between port reinitializations is set to 2 seconds.

## Parameters

*seconds*

Specifies the time between port reinitializations. The value can range from 1 through 10 seconds. The default is 2 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The LLDP re-initialization delay timer specifies the minimum number of seconds the device will wait from when LLDP is disabled on a port, until it will honor a request to re-enable LLDP on that port.

The **no** form of the command removes the configured value and restores the interval between port reinitializations to the default of 2 seconds.

## Examples

The following example sets the reinitialization delay timer to 5 seconds.

```
device(config)# lldp reinit-delay 5
```

# lldp run

Enables Link Layer Discovery Protocol (LLDP) globally.

## Syntax

**lldp run**

**no lldp run**

## Command Default

LLDP is disabled globally.

## Modes

Global configuration mode

## Usage Guidelines

To enable LLDP on individual ports, first LLDP must be enabled globally (on the entire device).

The **no** form of the command disables LLDP globally, with the exception of enabled SPX (802.1br) ports.

When the **spx cb-enable** command is entered, LLDP is automatically enabled on SPX ports (802.1br communication ports, not to be confused with PE unit data ports). Enabling or disabling LLDP on data ports has no impact on SPX ports. When the **no spx cb-enable** command is entered, SPX ports follow the LLDP global state; that is, if LLDP is disabled on data ports, it is also disabled on the SPX (802.1br) ports.

## Examples

The following example enables LLDP globally.

```
device(config)# lldp run
```

## History

| Release version | Command history |
|---|---|
| 8.0.40a | Global default behavior changes to disabled for data ports. SPX (802.1br) ports are enabled separately with the spx cb-enable command. |

# lldp snmp-notification-interval

Configures the minimum time between SNMP traps and syslog messages.

## Syntax

**lldp snmp-notification-interval** *seconds*

**no lldp snmp-notification-interval**

## Command Default

The default time between transmission of SNMP traps and syslog messages is 5 seconds.

## Parameters

*seconds*

Configures the time, in seconds, between transmission of SNMP traps and syslog messages. The value can range from 5 through 3600. The default is 5.

## Modes

Global configuration mode

## Usage Guidelines

When SNMP notifications and syslog messages for LLDP are enabled, the device will send no more than one SNMP notification and corresponding syslog message within a 5-second period.

The **no** form of the command removes the configured value and restores the time between transmission of SNMP traps and syslog messages to the default of 5 seconds.

## Examples

The following example sets the minimum time interval between traps and syslog messages to 60 seconds.

```
device(config)# lldp snmp-notification-interval 60
```

# lldp tagged-packets

Enables support for tagged Link Layer Discovery Protocol (LLDP) packets.

## Syntax

**lldp tagged-packets process**

**no lldp tagged-packets** [ **process** ]

## Command Default

By default, devices do not accept tagged LLDP packets from other vendor devices.

## Parameters

**process**

Enables processing of tagged LLDP packets.

## Modes

Global configuration mode

## Usage Guidelines

When support for tagged LLDP packets is enabled, the device accepts incoming LLDP tagged packets if the VLAN tag matches any of the following:

- A configured VLAN on the port
- The default VLAN for a tagged port
- The configured untagged VLAN for a dual-mode port

The **no** form of the command disables support for tagged LLDP packets.

## Examples

The following example enables support for tagged LLDP packets.

```
device(config)# lldp tagged-packets process
```

# lldp transmit-delay

Configures the minimum time between Link Layer Discovery Protocol (LLDP) transmissions.

## Syntax

**lldp transmit-delay** *seconds*

**no lldp transmit-delay**

## Command Default

When LLDP is enabled, the system automatically sets the LLDP transmit delay to 2 seconds.

## Parameters

*seconds*

Configures the LLDP transmit delay, in seconds. The value can range from 1 through 8192. The default value is 2.

## Modes

Global configuration mode

## Usage Guidelines

The LLDP transmit delay must not be greater than one quarter of the LLDP transmission interval (CLI command **lldp transmit-interval**).

The LLDP transmit delay prevents an LLDP agent from transmitting a series of successive LLDP frames during a short time period, when rapid changes occur in LLDP. It also increases the probability that multiple changes, rather than a single change, will be reported in each LLDP frame.

The **no** form of the command removes the configured value and restores the default value of 2 seconds.

## Examples

The following example sets the LLDP transmit delay to 7 seconds.

```
device(config)# lldp transmit-delay 7
```

# lldp transmit-hold

Configures the transmit holdtime multiplier for time to live (TTL).

## Syntax

**lldp transmit-hold** *value*

**no lldp transmit-hold** [ *value* ]

## Command Default

When LLDP is enabled, the device automatically sets the holdtime multiplier for TTL to 4.

## Parameters

*value*

Configures the transmit holdtime multiplier. The value can range from 2 to 10. The default is 4.

## Modes

Global configuration mode

## Usage Guidelines

The transmit holdtime multiplier for TTL is used to compute the actual TTL value used in an Link Layer Discovery Protocol (LLDP) frame. The TTL value is the length of time for which the receiving device maintains information in its MIB.

**NOTE**
Setting the transmit interval or transmit holdtime multiplier, or both, to inappropriate values can cause the LLDP agent to transmit LLDP PDUs with TTL values that are excessively high. This, in turn, can affect how long a receiving device retains information if it is not refreshed.

The **no** form of the command removes the configured value and restores the holdtime multiplier for TTL to the default value 4.

## Examples

The following example sets the holdtime multiplier to 6.

```
device(config)# lldp transmit-hold 6
```

# lldp transmit-interval

Sets the interval between regular Link Layer Discovery Protocol (LLDP) packet transmissions.

## Syntax

**lldp transmit-interval** *seconds*

**no lldp transmit-interval**

## Command Default

When LLDP is enabled, the transmit interval between LLDP packets is set to 30 seconds.

## Parameters

*seconds*

Configures the time interval, in seconds, between LLDP packet transmissions. The value can range from 5 through 32768.

## Modes

Global configuration mode

## Usage Guidelines

Setting the transmit interval or transmit holdtime multiplier, or both, to inappropriate values can cause the LLDP agent to transmit LLDP PDUs with TTL values that are excessively high. This in turn can affect how long a receiving device retains the information if it is not refreshed.

The **no** form of the command removes the configured value and sets the time interval between LLDP packet transmissions to 30 seconds.

## Examples

The following example sets the time interval between LLDP packet transmissions to 100 seconds.

```
device(config)# lldp transmit-interval 40
```

# load-balance symmetric

Enables symmetric load balancing for IPv4 and IPv6 data traffic on Ruckus FastIron devices.

## Syntax

**load-balance symmetric**

**no load-balance symmetric**

## Modes

Global configuration mode

## Usage Guidelines

This command configuration affects selection of LAG member port after symmetric load balancing is enabled. For a bidirectional (forward and reverse direction) traffic flow, same port in the LAG and/or same next hop for ECMP is chosen.

The **no** form of the command disables symmetric load balancing in the system.

## Examples

The following example enables symmetric load balancing for IPv4 and IPv6 data traffic on a device.

```
device(config)# load-balance symmetric
```

## History

| Release version | Command history |
|---|---|
| 8.0.30b | This command was introduced. |

# local-as

Specifies the BGP autonomous system number (ASN) where the device resides.

## Syntax

**local-as** *num*

**no local-as** *num*

## Parameters

*num*

The local ASN. The range is from 1 through 4294967295.

## Modes

BGP configuration mode

## Usage Guidelines

Use the **no** form of this command to remove the ASN from the device.

ASNs in the range from 64512 through 65535 are private numbers that are not advertised to the external community.

## Examples

This example assigns a separate local AS number.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# local-as 777
```

# local-certificate (PKI)

Defines te URL of the local certificate.

## Syntax

**local-certificate** { *url* }

**no local-certificate** { *url* }

## Command Default

## Parameters

*url*

Defines the location where local certificates are stored.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

## Examples

The following example configures the storage location for location certificates as the URL shown.

```
http://FI-PKI02.englab.ruckus.com/CertSrv/localcert.pem
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# local-identifier

Configures the local system identifier for an Internet Key Exchange version 2 (IKEv2) profile.

## Syntax

**local-identifier** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

**no local-identifier** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

## Command Default

The device IP address is used as the local identifier.

## Parameters

**address** *ip-address*

Specifies an address as the local system identifier.

*ip-address*

Specifies an IPv4 address.

*ipv6-address*

Specifies an IPv6 address.

**dn** *dn-name*

Specifies a Distinguished Name (DN) as the local system identifier.

**email** *email-address*

Specifies an email address as the local system identifier.

**fqdn** *fqdn-name*

Specifies a fully qualified domain name (FQDN) as the local system identifier.

**key-id** *key-id*

Specifies a key ID as the local system identifier.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

The **no** form of the command removes the specified local identifier.

# Examples

The following example shows how to configure IP address 10.3.3.3 as the local system identifier for an IKEv2 profile named prof_mktg.

```
device# configure terminal
device(config)# ikev2 profile prof-mktg
device(config-ike-profile-prof-mktg)# local-identifier address 10.3.3.3
device(config-ike-profile-prof-mktg)# exit
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support was added for IPv6. |

# local-userdb

Creates a local user database.

## Syntax

**local-userdb** *db-name*

**no local-userdb** *db-name*

## Command Default

No local user databases exists.

## Parameters

*db-name*

Configures the name of the local user database. The name can be up to 31 alphanumeric characters.

## Modes

Global configuration mode

## Usage Guidelines

Ruckus supports a maximum of ten local user databases, each containing up to 50 user records. Each user record consists of a username and password.

The **no** form of the command removes a local database.

## Examples

The following example shows how to configure a local user database.

```
device(config)# local-userdb userdb1
device(config-localuserdb-userdb1)#
```

# log (OSPFv2)

Controls the generation of OSPFv2 logs.

## Syntax

**log** { **adjacency** [ **dr-only** ] | **all** | **bad_packet** [ **checksum** ] | **database** | **memory** | **retransmit** }

**no log** { **adjacency** [ **dr-only** ] | **all** | **bad_packet** [ **checksum** ] | **database** | **memory** | **retransmit** }

## Command Default

Only OSPFv2 messages indicating possible system errors are logged. Refer to the Parameters section for specific defaults.

## Parameters

**adjacency**

Specifies the logging of essential OSPFv2 neighbor state changes. This option is disabled by default.

**dr-only**

Specifies the logging of essential OSPF neighbor state changes where the interface state is designated router (DR).

**all**

Specifies the logging of all syslog messages.

**bad-packet**

Specifies the logging of bad OSPFv2 packets. This option is enabled by default.

**checksum**

Specifies all OSPFv2 packets that have checksum errors.

**database**

Specifies the logging of OSPFv2 LSA-related information. This option is disabled by default.

**memory**

Specifies the logging of OSPFv2 memory issues. This option is enabled by default.

**retransmit**

Specifies the logging of OSPFv2 retransmission activities. This option is disabled by default.

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

Use this command to disable or re-enable the logging of specific events related to OSPFv2. If this command is not enabled only OSPFv2 messages indicating possible system errors are logged.

For interfaces where the designated router state is not applicable, such as point-to-point and virtual links, OSPF neighbor state changes are always logged irrespective of the setting of the **dr-only** sub-option.

A limitation with the **dr-only** sub-option is that when a DR/BDR election is underway, OSPF neighbor state changes pertaining to non-DR/BDR routers are not logged. Logging resumes once a DR is elected on that network.

The **no** form of the command restores the default settings. Use the **no log all** command to return all OSPFv2 logging options to the default settings.

# Examples

The following example enables the logging of all OSPFv2-related syslog events.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# log all
```

The following example enables the logging of OSPFv2 retransmission activities.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# log retransmit
```

# logging

Enables logging on the Router Advertisement (RA) guard policy.

## Syntax

**logging**

**no logging**

## Modes

RA guard policy configuration mode

## Usage Guidelines

The **no** form of this command disables logging on the policy.

Logging cannot be modified if the RA guard policy is in use.

You can verify the logs for RA guard, such as RAs dropped, permitted, count for dropped packets, and reasons for the drop.

Logging increases the CPU load and for higher traffic rates, RA packets drop due to congestion if they are received at the line rate. For less load on the CPU, logging can be disabled on the RA guard policy.

## Examples

The following example enables logging on an RA guard policy:

```
device(config)# ipv6 raguard policy p1
device(config-ipv6-RAG-policy p1)# logging
```

# logging buffered

Enables logging of specific messages or changes the number of entries that the local syslog buffer can store.

## Syntax

**logging buffered** { *level* | *num-entries* }

**no logging buffered** { *level* | *num-entries* }

## Command Default

The number of entries that the local syslog buffer can store is 4000.

## Parameters

*level*

Specifies the message level with one of the following values: **alerts**, **critical**, **debugging**, **emergencies**, **errors**, **informational**, **notifications**, **warnings**.

*num-entries*

Configures the number of entries that the local syslog buffer can store. The value ranges from 1 through 4000.

## Modes

Global configuration mode

## Usage Guidelines

The software does not log informational or debugging messages.

To change the message level, you must disable logging of specific message levels individually.

When changing the number of entries that the local syslog buffer can store, pay attention to the following considerations:

- You must save the configuration and reload the software to effect the change.
- The modified number of syslog messages remains persistent across reloads if the **logging persistence** command is configured.
- The number of persistent log messages across soft reboots is the same as the number of dynamic syslog messages.
- If you decrease the size of the buffer, the software clears the buffer before effecting any changes.
- If you increase the size of the syslog buffer, the software clears some of the older locally buffered syslog messages.

The **no** form of the command using the *level* option disables logging of the specified message level. The **no** form of the command using the *num-entries* option resets the syslog buffer size to 4000 (the default).

# Examples

The following example enables the logging of debugging messages.

```
device(config)# logging buffered debugging
```

The following example sets the number of entries that the local syslog buffer can store to 1000.

```
device(config)# logging buffered 1000
```

The modified number of dynamic syslog messages getting logged is displayed in the **show logging** command output.

```
device# show logging
Syslog logging: enabled ( 0 messages dropped, 0 flushes, 0 overruns)
Buffer logging: level ACDMEINW, 9 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
Dec 20 03:51:04:I:System: Stack unit 1   Power supply 1  is up

Dynamic Log Buffer (1000 lines):
Dec 20 03:51:35:I:Security: console login by un-authenticated console user to PRIVILEGED EXEC mode
Dec 20 03:51:04:I:System: Stack unit 1   Power supply 1  is up
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was modified to increase the default value of dynamic syslog messages being logged from 50 to 4000. |

# logging console

Enables the real-time display of syslog messages.

## Syntax

**logging console**

**no logging console**

## Command Default

To view syslog messages generated by a device, you need to display the syslog buffer or the log on a syslog server used by the device.

## Modes

Global configuration mode

## Usage Guidelines

You can enter this command from the serial console or from a Telnet or SSH session.

You can enable the real-time display of syslog messages on the management console. When you enable this command, the software displays a syslog message on the management console when the message is generated. However, to enable the display of real-time syslog messages in Telnet or SSH sessions, you must also enable the display within the individual sessions.

The **no** form of the command disables the real-time display of syslog messages.

## Examples

The following example enables the real-time display of syslog messages.

```
device(config)# logging console
```

# logging cli-command

Enables logging of all syntactically valid CLI commands from each user session into the system log.

## Syntax

**logging cli-command**

**no logging cli-command**

## Command Default

Logging of CLI commands is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

If the **logging cli-command** command is configured, all the CLI commands executed by the user are logged in the system log and are displayed in the **show logging** command output.

The **no** form of the command disables the logging of CLI commands from each user session into the system log.

## Examples

The following example enables the logging of CLI commands on the device.

```
device(config)# logging cli-command
```

The following example shows the system log records which are displayed in the **show logging** command output. The system log contains the valid commands that are executed by the user.

```
device (config)# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 5 overruns)
            Buffer logging: level ACDMEINW, 50 messages logged
            level code: A=alert C=critical D=debugging M=emergency E=error I=informational
N=notification W=warning
Dynamic Log Buffer (50 lines):
8d02h28m43s:I:CLI CMD: "ip route 0.0.0.0 0.0.0.0 10.20.64.1" by un-authenticated
user from console
8d02h28m24s:I:System: Interface ethernet 1/1/1, state up
8d02h28m22s:I:CLI CMD: "enable" by un-authenticated user from console
8d02h28m22s:I:PORT: 1/1/1 enabled by un-authenticated user from console session
8d02h28m19s:I:CLI CMD: "disable" by un-authenticated user from console
8d02h28m19s:I:PORT: 1/1/1 disabled by un-authenticated user from console session
8d02h28m16s:I:CLI CMD: "interface ethernet 1/1/1" by un-authenticated user from
console
```

# logging-enable

Enables logging within a specified IPv6 access control list (ACL) for rules that include the **log** keyword.

## Syntax

**logging-enable**

**no logging-enable**

## Command Default

IPv6 ACL logging is not enabled.

## Modes

IPv6 ACL configuration mode

## Usage Guidelines

The **log** parameter is effective in both deny rules and permit rules.

The **log** parameter is effective in both ingress and egress ACLs.

This command enables logging for IPv6 ACLs. For IPv4 ACLs, refer to the **acl-logging** command topic.

The **no** form of the command disables logging for the specified IPv6 ACL.

## Examples

The following example enables logging within a specified IPv6 ACL for rules that include the **log** keyword. It then applies that ACL to an interface to filter incoming traffic.

```
device# configure terminal
device(config)# ipv6 access-list ACL6_log
device(config-ipv6-access-list ACL6_log)# logging-enable
device(config-ipv6-access-list ACL6_log)# remark The following rule permits ipv6 packets from
2001:DB8::2 to any destination
device(config-ipv6-access-list ACL6_log)# permit ipv6 host 2001:DB8::2 any log
device(config-ipv6-access-list ACL6_log)# remark The following rule denies udp packets from any source
to any destination
device(config-ipv6-access-list ACL6_log)# deny udp any any log
device(config-ipv6-access-list ACL6_log)# remark The following rule denies IPv6 packets from any source
to any destination
device(config-ipv6-access-list ACL6_log)# deny ipv6 any any
device(config-ipv6-access-list ACL6_log)# interface ethernet 1/9/12
device(config-if-e1000-1/9/12)#ipv6 traffic-filter ACL_log_v6 in
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was modified to support permit rules (in addition to deny rules) and egress ACLs (in addition to ingress ACLs). |

# logging enable config-changed

Configures a device to generate syslog messages when the startup-config file is changed.

## Syntax

**logging enable config-changed**

**no logging enable config-changed**

## Command Default

The trap is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables the generation of the syslog messages when the startup-config file is changed.

## Examples

The following example enables syslog messages when the startup-config file is changed.

```
device(config)# logging enable config-changed
```

# logging enable ikev2

Enables system log messages and traps for IKEv2 events.

## Syntax

**logging enable ikev2** [ **ikev2-packet** | **ikev2-extended** ]

**no logging enable ikev2** [ **ikev2-packet** | **ikev2-extended** ]

## Command Default

Log messages for IKEv2 events are enabled. Log messages for IKEv2 extended events and IKEv2 packets are not enabled.

## Parameters

**ikev2-packet**
Specifies logging of IKEv2 packets.

**ikev2-extended**
Specifies logging of IKEv2 extended events.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command disables the generation of the specified syslog messages and traps.

This command is supported on the Ruckus ICX 7450, with an FPGA-based add-on crypto card.

## Examples

The following example configures syslog generation for IKEv2 events.

```
device# configure terminal
device(config)# logging enable ikev2
```

The following example configures logging of additional IKEv2 events.

```
device# configure terminal
device(config)# logging enable ikev2 ikev2-extended
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | This command was modified to include logging options for IKEv2 extended events and packets. |

# logging enable ipsec

Enables system log messages and traps for IPsec events.

## Syntax

**logging enable ipsec**

**no logging enable ipsec**

## Command Default

Log messages for IPsec events are enabled.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command disables the generation of the specified syslog messages and traps.

This command is supported on the Ruckus ICX 7450, with an FPGA-based add-on crypto card.

## Examples

The following example configures syslog generation for IPsec events.

```
device(config)# logging enable ipsec
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced |

# logging enable (PKI)

Enables logging of PKI events and, as an option, PKI packets.

## Syntax

**logging enable** { **pki** | **pki-extended** }

**no logging enabe** { **pki** | **pki-extended** }

## Command Default

PKI events and packets are not logged by default.

## Parameters

**pki**

Specifies that PKI events be logged.

**pki-extended**

Specifies that PKI events and packets be logged.

## Modes

Global configuration mode.

## Usage Guidelines

The no form of the command disables PKI logging or PKI extended logging.

## Examples

The following example enables logging of PKI events and packets.

```
device# configure terminal
device(config)# enable logging pki-extended
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# logging enable rfc5424

Enables Syslog logging in accordance with RFC 5424 which provides the maximum amount of information in every Syslog in a structured format.

## Syntax

**logging enable rfc5424**

**no logging enable rfc5424**

## Command Default

Syslog is generated in accordance with RFC 3164.

## Modes

Global configuration mode

## Usage Guidelines

The Logging buffer must be cleared before enabling Syslog specific to RFC 5424, otherwise system throws an error.

If the **logging cli-command** command is present in the running configuration, switching between the default RFC 3164 Syslog logging and the RFC 5424-specific Syslog logging is not supported.

The **no** form of the command enables Syslog logging in accordance with RFC 3164.

## Examples

The following example enables Syslog logging in accordance with RFC 5424.

```
device(config)# clear logging
device(config)# logging enable rfc5424
```

The following example removes the configuration to enable Syslog logging specific to RFC 5424 and enables Syslog logging in accordance with RFC 3164.

```
device(config)# clear logging
device(config)# no logging enable rfc5424
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 08.0.30h | Support for the command was added. |

# logging enable user-login

Enables viewing user login details in syslog messages and traps.

## Syntax

**logging enable user-login**

**no logging enable user-login**

## Command Default

User login details in syslog messages and traps are not enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

Ruckus devices send syslog messages and SNMP traps when a user logs in to or out of user EXEC or privileged EXEC mode in the CLI. The feature applies to users whose access is authenticated by an authentication method list based on a local user account, RADIUS server, or TACACS/TACACS+ server.

The **no** form of the command disables the user login details from syslog messages and traps.

## Examples

The following example enables viewing the user login details.

```
device(config)# logging enable user-login
```

# logging facility

Configures the log facility to log messages from the device.

## Syntax

**logging facility** *facility-name*

**no logging facility**

## Command Default

The default facility for messages that the device sends to the syslog server is "user".

## Parameters

*facility-name*

Specifies the facility to log the messages from the device. The facility name can be one of the following:

**auth**

Security or authorization messages.

**cron**

cron/at subsystem.

**daemon**

System daemons.

**kern**

Kernel messages.

**local0**

Reserved for local use.

**local1**

Reserved for local use.

**local2**

Reserved for local use.

**local3**

Reserved for local use.

**local4**

Reserved for local use.

**local5**

Reserved for local use.

**local6**

Reserved for local use.

**local7**

Reserved for local use.

**lpr**
Line printer subsystem.

**mail**
Mail system.

**news**
Netnews subsystem.

**syslog**
Messages generated internally by syslog.

**sys9**
Reserved for system use.

**sys10**
Reserved for system use.

**sys11**
Reserved for system use.

**sys12**
Reserved for system use.

**sys13**
Reserved for system use.

**sys14**
Reserved for system use.

**user**
Random user-level messages.

**uucp**
UUCP subsystem.

# Modes

Global configuration mode

# Usage Guidelines

The syslog daemon on the syslog server uses a facility to determine where to log the messages from the Ruckus device. You can specify only one facility. If you configure the device to use two syslog servers, the device uses the same facility on both servers.

The **no** form of the command restores the default facility.

# Examples

The following example changes the log facility.

```
device(config)# logging facility local0
```

# logging host

Configures a syslog server.

## Syntax

**logging host** { *ipv4-addr* | *server-name* | **ipv6** *ipv6-addr* } [ **udp-port** *number* ]

**no logging host** { *ipv4-addr* | *server-name* | **ipv6** *ipv6-addr* } [ **udp-port** *number* ]

## Command Default

Syslog server is not configured.

## Parameters

*ipv4-addr*
Configures a syslog server with the specified IPv4 address.

*server-name*
Configures a syslog server with the specified name..

**ipv6** *ipv6-addr*
Configures a syslog server with the specified IPv6 address.

**udp-port** *number*
Specifies the UDP port number.

## Modes

Global configuration mode

## Usage Guidelines

You can specify up to six syslog servers by configuring the command.

The **no** form of the command removes the syslog server configuration.

## Examples

The following example configures a syslog server with IP address 10.0.0.99.

```
device(config)# logging host 10.0.0.99
```

To specify an additional syslog server, enter the **logging host** command again.

```
device(config)# logging host 10.0.0.100
```

# logging on

Enables local syslog logging.

## Syntax

**logging on**

**no logging on**

## Command Default

Local syslog logging is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

This command enables local syslog logging with the following defaults:

- Messages of all severity levels (Emergencies through Debugging) are logged.
- Up to 50 messages are retained in the local syslog buffer.
- No syslog server is specified.

The **no** form of the command disables local syslog logging.

## Examples

The following example enables local syslog logging.

```
device(config)# logging on
```

# logging persistence

Configures the device to save system log messages after a soft reboot.

## Syntax

**logging persistence**

**no logging persistence**

## Command Default

Logging persistence is not configured.

## Modes

Global configuration mode

## Usage Guidelines

If the syslog buffer size was set to a different value using the command **logging buffered**, the system log will be cleared after a soft reboot, even if this feature is enabled. This clearing will occur only with a soft reboot immediately following a syslog buffer size change. A soft reboot by itself will not clear the system log. To prevent the system from clearing the system log, leave the number of entries allowed in the syslog buffer unchanged.

Enabling logging persistence does not save syslog messages after a hard reboot. When the device is power-cycled, the syslog messages are cleared.

If logging persistence is enabled and you load a new software image on the device, you must first clear the log if you want to reload the device.

The **no** form of the command disables the device from saving system log messages after a soft reboot.

## Examples

The following example configures the device to save system log messages after a soft reboot.

```
device(config)# logging persistence
```

# login-page

Configures the login page details to redirect the client to the login page hosted on the external captive portal server.

## Syntax

Syntax for use with a Ruckus Cloudpath server:
**login-page /enroll/** *page-name*

Syntax for use with an Aruba Clearpass server:
**login-page /guest/** *page-name*

Syntax for use with an Cisco ISE server:
**login-page** *page-name*

**no login-page** *page-name*

## Command Default

Login page for redirecting the client is not configured.

## Parameters

*page-name*

Specifies the login page created on the external captive portal server. For Cisco ISE servers, the name of the login page is created by the server.

## Modes

Captive portal configuration mode

## Usage Guidelines

Note that the terms Captive Portal and external web authentication are used interchangeably.

The login page details must be same as the login page hosted on the external captive portal server.

The **no** form of the command removes the login page configuration.

## Examples

The following example configures the login page details to redirect the client to the login page hosted on the external captive portal server, which in this case is a Ruckus Cloudpath server.

```
device(config)# captive-portal cp_ruckus
device(config-cp-cp_ruckus)# login-page /enroll/ruckusguestlogin.php
```

The following example configures the login page details when an Aruba Clearpass server is used.

```
device(config)# captive-portal cp_ruckus
device(config-cp-cp_ruckus)# login-page /guest/ruckus/guestlogin.php
```

The following example configures the login page details when a Cisco ISE server is used.

```
device(config)# captive-portal cp_ruckus
device(config-cp-cp_ruckus)# login-page ruckusguestlogin.php
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j |

# log-status-change

Controls the generation of all OSPFv3 logs.

## Syntax

**log-status-change**

**no log-status-change**

## Command Default

Disabled

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

Use this command to disable or re-enable the logging of events related to OSPFv3, such as neighbor state changes and database overflow conditions.

The **no** form of this command disables the logging of events.

## Examples

The following example disables the logging of events.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# no log-status-change
```

The following example enables the logging of events.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# log-status-change
```

# loop-detection

Enables loop detection on a physical port (Strict Mode) or on a VLAN (Loose Mode).

## Syntax

**loop-detection** [ **shutdown-disable** ]

**no loop-detection** [ **shutdown-disable** ]

## Command Default

Loop detection is disabled by default.

## Parameters

**shutdown-disable**
Disables shutdown of a port due to loop detection.

## Modes

Interface configuration mode

VLAN configuration mode

## Usage Guidelines

By default, the port sends test packets every one second, or the number of seconds specified by the **loop-detection-interval** command.

The **no** form of the command disables loop detection.

## Examples

The following example enables loop detection on a physical port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# loop-detection
```

The following example enables loop detection on a VLAN.

```
device(config)# vlan 20
device(config-vlan-20)# loop-detection
```

# loop-detection-interval

Configures how often a test packet is sent on a port.

## Syntax

**loop-detection-interval** *number*

**no loop-detection-interval** *number*

## Command Default

Loop detection time is set to 1 second.

## Parameters

*number*

Specifies a value from 1 to 100 seconds. The system multiplies the entry by 0.1 to calculate the interval at which test packets are sent.

## Modes

Global configuration mode

## Usage Guidelines

When loop detection is enabled, the loop-detection time unit is 0.1 second, with a default of 10 (1 second). The range is from 1 (one tenth of a second) to 100 (10 seconds). You can use the **show loop-detection status** command to view the loop-detection interval.

The **no** form of the command sets the loop detection interval to the default global loop-detection interval of 1 second.

## Examples

The following example sets the loop-detection interval to 5 seconds (50*0.1).

```
device(config)# loop-detection-interval 50
```

# loop-detection shutdown-disable

Disables shutdown of a port when a loop detection probe packet is received on an interface.

## Syntax

**loop-detection shutdown-disable**

**no loop-detection shutdown-disable**

## Command Default

Loop detection shutdown is enabled on the interface.

## Modes

Interface configuration

## Usage Guidelines

The **no** form of this command disables loop detection shutdown.

Shutdown prevention for loop-detect functionality allows users to disable shut down of a port when the loop detection probe packet is received on an interface. This provides control over deciding which port is allowed to enter in to an error-disabled state and go into a shutdown state when a loop is detected.

## Examples

The following example disables loop detection shutdown on an interface.

```
device(config)# interface ethernet 1/1/7
device(config-if-e1000-1/1/7)# loop-detection shutdown-disable
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# loop-detection-syslog-interval

Specifies the interval (in minutes) at which a syslog is generated.

## Syntax

**loop-detection-syslog-interval** *num*

**no loop-detection-syslog-interval** *num*

## Command Default

The syslog interval is 5 minutes.

## Parameters

*num*

Specifies the syslog interval in minutes. The interval can range from 1 through 1440 minutes.

## Modes

Global configuration

## Usage Guidelines

The **no** form of this command restores the default settings.

You can specify the interval at which the loop detection syslog message is generated if the **loop-detection-shutdown-disable** command is configured for the port. This configuration applies to all the ports that have loop detection shutdown prevention configured.

## Examples

The following example shows the loop detection syslog interval set to 1 hour.

```
device(config)# loop-detection-syslog-interval 60
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# l2protocol dot1q-tunnel

Enables Q-in-Q BPDU tunneling on an interface.

## Syntax

**l2protocol dot1q-tunnel** [ **cdp** | **lacp** | **lldp** | **stp** ]

**no l2protocol dot1q-tunnel** [ **cdp** | **lacp** | **lldp** | **stp** ]

## Parameters

**cdp**
Enables CDP tunneling on the interface.

**lacp**
Enables LACP tunneling on the interface.

**lldp**
Enables LLDP tunneling on the interface.

**stp**
Enables STP (PVST, STP, RSTP, MSTP) tunneling on the interface.

## Modes

Interface configuration mode

## Usage Guidelines

When Q-in-Q BPDU tunneling is enabled on customer connected interface of the service provider device, all the received tunnel protocol packets will be tunneled to the service network. To prevent any locally generated (STP, LLDP, and so on) protocol packets on the service provider network from switching to the customer side, the corresponding protocols must be disabled on the device.

Q-in-Q BPDU tunneling is supported on tag-profile enabled port or selective Q-in-Q enabled port.

The **no** form of the command disables Q-in-Q tunneling.

## Examples

The following example shows how to enable Q-in-Q BPDU tunneling on a tag-profile enabled port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# tag-profile enable
device(config-if-e1000-1/1/1)# l2protocol dot1q-tunnel
```

The following example shows how to enable Q-in-Q BPDU tunneling on a selective Q-in-Q enabled port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# qinq-tunnel cvlan 1 to 4 untag svlan 100
device(config-if-e1000-1/1/1)# l2protocol dot1q-tunnel
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# l2protocol dot1q-tunnel cos

Specifies a global Class of Service (CoS) value for Q-in-Q tunnel ports.

## Syntax

**l2protocol dot1q-tunnel cos** *cos-value*

**no l2protocol dot1q-tunnel cos** *cos-value*

## Command Default

The default CoS value is 5.

## Parameters

*cos-value*

Specifies the CoS value globally so that the ingress BPDUs on the tunnel ports are encapsulated with the specified class. Valid values are from 1 through 7.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the CoS value to the default value.

## Examples

The following example sets the global CoS value for Q-in-Q BPDU tunneling as 6.

```
device# configure terminal
device(config)# l2protocol dot1q-tunnel cos 6
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# l2protocol dot1q-tunnel drop-threshold

Configures the maximum number of packets that can be processed on an interface for Q-in-Q BPDU tunneling before being dropped.

## Syntax

**l2protocol dot1q-tunnel drop-threshold** { **all** | **cdp** | **lacp** | **lldp** | **stp** } *threshold-value*

**no l2protocol dot1q-tunnel drop-threshold** { **all** | **cdp** | **lacp** | **lldp** | **stp** } *threshold-value*

## Parameters

**all**
Configures the drop threshold for all protocol packets.

**cdp**
Configures the drop threshold for CDP packets.

**lacp**
Configures the drop threshold for LACP packets.

**lldp**
Configures the drop threshold for LLDP packets.

**stp**
Configures the drop threshold for STP packets.

*threshold-value*
Specifies the threshold rate in packets per second. Valid values are from are from 1 through 4096.

## Modes

Interface configuration mode

## Usage Guidelines

When the **all** option is used on an interface, individual protocol option cannot be used.

The **no** form of the command resets the threshold values to 0 and disables the drop threshold.

## Examples

The following example configures the port drop threshold value for a Q-in-Q tunneling port.

```
device(config)# interface ethernet 1/1/1
(config-if-e1000-1/1/1)# l2protocol dot1q-tunnel drop-threshold all 3000
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# l2protocol dot1q-tunnel-mac

Specifies the multicast MAC address for Q-in-Q BPDU tunneling.

## Syntax

**l2protocol dot1q-tunnel-mac** { *mac-address* | **original** }

**no l2protocol dot1q-tunnel-mac** { *mac-address* | **original** }

## Command Default

The default MAC value is 0100.0ccd.cdd1.

## Parameters

*mac-address*
Specifies the tunnel MAC address.

**original**
Specifies to use the original MAC address in the packet as tunnel MAC address.

## Modes

Global configuration mode

## Usage Guidelines

The original MAC address can be used as an option to interoperate with older ICX 6xxx series devices.

The **original** option must not be used if the transit switch in the service provider network is an ICX 6xxx device or any vendor that consumes standard BPDU as it may result in protocol packet drop.

Both Server Provider Edge switches must have the same tunnel MAC address.

The **no** form of the command sets the multicast MAC address for Q-in-Q BPDU tunneling to the default value.

## Examples

The following example sets the multicast MAC address for Q-in-Q BPDU tunneling.

```
device# configure terminal
device(config)# l2protocol dot1q-tunnel-mac 0100.1a2b.3c4d
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# l2protocol dot1q-tunnel shutdown-threshold

Specifies the maximum number of packets that can be processed on an interface for Q-in-Q BPDU tunneling beyond which the ingress port is put in error-disabled state.

## Syntax

**l2protocol dot1q-tunnel shutdown-threshold** { **all** | **cdp** | **lacp** | **lldp** | **stp** } *threshold-value*

**no l2protocol dot1q-tunnel shutdown-threshold** { **all** | **cdp** | **lacp** | **lldp** | **stp** } *threshold-value*

## Parameters

**all**
Configures the shutdown threshold for all protocol packets.

**cdp**
Configures the shutdown threshold for CDP packets.

**lacp**
Configures the shutdown threshold for LACP packets.

**lldp**
Configures the shutdown threshold for LLDP packets.

**stp**
Configures the shutdown threshold for STP packets.

*threshold-value*
Specifies the threshold rate in packets per second. Valid values are from are from 1 through 4096.

## Modes

Interface configuration mode

## Usage Guidelines

When the **all** option is used on an interface, individual protocol option cannot be used.

The **no** form of the command resets the threshold values to 0 and disables the shutdown threshold.

## Examples

The following example configures the port shutdown threshold value for a Q-in-Q tunneling port.

```
device(config)# interface ethernet 1/1/1
(config-if-e1000-1/1/1)# l2protocol dot1q-tunnel shutdown-threshold all 3000
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# Commands M

## mac-age-time

Configures the MAC address age timer.

### Syntax

**mac-age-time** *seconds*

**no mac-age-time** *seconds*

### Command Default

The default MAC address age timeout is 300 seconds.

### Parameters

*seconds*

Timeout value in seconds.The timeout value range is 0 (disabled) or from 10 through 86,400 seconds.

### Modes

Global configuration mode

### Usage Guidelines

To disable the MAC address age timer, set the timeout value to 0.

If the total number of MAC addresses in the system is more than 16, 000, Ruckus recommends a MAC address age timer greater than 60 seconds. If the total number of MAC addresses in the system is more than 64, 000, Ruckus recommends a MAC address age timer greater than 120 seconds.

Usually, the actual MAC address age time is from one to two times the configured value. For example, if you set the MAC address age timer to 60 seconds, learned MAC address entries age out after remaining unused for between 60 and 120 seconds. However, if all of the following conditions are met, then the MAC address entries age out after a longer than expected duration:

- The MAC address age timer is set to greater than 630 seconds.
- The number of MAC address entries is over 6, 000.
- All MAC address entries are learned from the same packet processor.
- All MAC address entries age out at the same time.

The **no** form of the command resets the MAC address age timeout value to the default value.

# Examples

The following example configures the MAC address age timeout to 570 seconds.

```
device(config)# mac-age-time 570
```

# mac-authentication auth-filter

Applies the specified filter on the interface and the MAC addresses defined in the filter (MAC filter) do not have to go through authentication.

## Syntax

**mac-authentication auth-filter** *filter-id vlan-id*

**no mac-authentication auth-filter** *filter-id vlan-id*

## Command Default

There are no filters applied on the interface.

## Parameters

*filter-id*

Specifies the identification number of the filter to be applied on the interface.

*vlan-id*

Specifies the identification number of the VLAN to which the filter is applied.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command disables this functionality.

A client can be authenticated in an untagged VLAN or tagged VLAN using the MAC address filter for MAC authentication. If auth-filter has tagged VLAN configuration, the clients are authenticated in auth-default VLAN and tagged VLAN provided in auth-filter. The clients authorized in auth-default VLAN allow both untagged and tagged traffic.

If the VLAN is not specified in the command, the auth-default VLAN is used.

## Examples

The following example applies the MAC address filter on VLAN 2.

```
device(config)# authentication
device(config-authen)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# mac-auth auth-filter 1 2
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# mac-authentication dot1x-disable

Configures the device so that 802.1X authentication is not performed after successful MAC authentication.

## Syntax

**mac-authentication dot1x-disable**

**no mac-authentication dot1x-disable**

## Command Default

By default, 802.1X authentication is also performed after successful MAC authentication.

## Modes

Authentication configuration sub-mode.

## Usage Guidelines

The **no** form of the command restores the default.

The command is applicable only when the authentication sequence is configured as MAC authentication followed by 802.1X authentication.

## Examples

The following example disables 802.1X authentication following a successful MAC authentication when MAC authentication is performed first.

```
device# configure terminal
device)config)# authentication
device(config-authen)# mac-authentication dot1x-disable
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# mac-authentication dot1x-override

Configures the device to perform 802.1X authentication when MAC authentication fails when the authentication sequence is configured as MAC authentication followed by 802.1X authentication.

## Syntax

**mac-authentication dot1x-override**

**no mac-authentication dot1x-override**

## Command Default

802.1X authentication is not performed when MAC authentication fails.

## Modes

Authentication configuration mode

## Usage Guidelines

This command is applicable only when the authentication sequence is configured as MAC authentication followed by 802.1X authentication.

If the **mac-authentication dot1x-override** command is configured, the clients that failed MAC authentication undergoes 802.1X authentication if the failure action is configured as restricted VLAN.

The **no** form of the command disables MAC authentication dot1x override functionality.

## Examples

The following example enables MAC authentication dot1x override when MAC authentication fails.

```
device(config)# authentication
device(config-authen)# mac-authentication dot1x-override
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# mac-authentication enable (Flexible authentication)

Enables MAC authentication globally or on a specific interface.

## Syntax

**mac-authentication enable** [ **all** | **ethernet** *device*/*slot*/*port* ]

**no mac-authentication enable** [ **all** | **ethernet** *device*/*slot*/*port* ]

## Command Default

MAC authentication is not enabled.

## Parameters

**all**

Enables MAC authentication on all interfaces.

**ethernet** *device*/*slot*/*port*

Enables MAC authentication on a specific interface.

## Modes

Authentication configuration mode

## Usage Guidelines

The **mac-authentication enable** command without any options initializes MAC authentication feature globally. The **mac-authentication enable** command with the **all** or **ethernet** options, enables MAC authentication on all or a specific interface respectively. After initializing MAC authentication feature using the **mac-authentication enable** command, you must enable MAC authentication on all or a specific interface.

The **no** form of the command disables MAC authentication.

## Examples

The following example globally enables MAC authentication.

```
device(config)# authentication
device(config-authen)# mac-authentcation enable
device(config-authen)# mac-authentication enable all
```

The following example enables MAC authentication on an interface.

```
device(config)# authentication
device(config-authen)# mac-authentcation enable
device(config-authen)# mac-authentication enable ethernet 1/1/11
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# mac-authentication password-format

Configures the MAC authentication password format.

## Syntax

**mac-authentication password-format** { **xx-xx-xx-xx-xx-xx** | **xx:xx:xx:xx:xx:xx** | **xxxx.xxxx.xxxx** | **xxxxxxxxxxxx** } [ **upper-case** ]

**no mac-authentication password-format** { **xx-xx-xx-xx-xx-xx** | **xx:xx:xx:xx:xx:xx** | **xxxx.xxxx.xxxx** | **xxxxxxxxxxxx** } [ **upper-case** ]

## Command Default

By default, the MAC address is sent to the RADIUS server in the format xxxxxxxxxxxx in lower case.

## Parameters

**xx-xx-xx-xx-xx-xx**
Specifies the MAC authentication password format as xx-xx-xx-xx-xx-xx.

**xx:xx:xx:xx:xx:xx**
Specifies the MAC authentication password format as xx:xx:xx:xx:xx:xx.

**xxxx.xxxx.xxxx**
Specifies the MAC authentication password format as xxxx.xxxx.xxxx.

**xxxxxxxxxxxx**
Specifies the MAC authentication password format as xxxxxxxxxxxx.

**upper-case**
Converts the password to uppercase.

## Modes

Authentication configuration mode

## Usage Guidelines

The **no** form of the command restores the default.

You can configure the device to send the MAC address to the RADIUS server in the format xx-xx-xx-xx-xx-xx, xx.xx.xx.xx.xx.xx, xxxx.xxxx.xxxx, or xxxxxxxxxxxx. Use the **upper-case** password format option to send the password in uppercase.

## Examples

The following example configures the MAC authentication password format as xx-xx-xx-xx-xx-xx.

```
device(config)# authentication
device(config-authen)# mac-authentication password-format xx-xx-xx-xx-xx-xx
```

The following example configures the MAC authentication password format as xx-xx-xx-xx-xx-xx in upper case.

```
device(config)# authentication
device(config-authen)# mac-authentication password-format xx-xx-xx-xx-xx-xx upper-case
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.20c | The **upper-case** option was added. |
| 08.0.61 | The **xx.xx.xx.xx.xx.xx** and the **xx:xx:xx:xx:xx:xx** options were added. |

# mac-authentication password-override (Flexible authentication)

Enables password override for MAC authentication and specifies a user-defined password instead of the MAC address for MAC authentication.

## Syntax

**mac-authentication password-override** *password*

**no mac-authentication password-override** *password*

## Command Default

MAC authentication password override is not enabled.

## Parameters

*password*

Specifies the password to be used for MAC authentication. The password can contain up to 32 alphanumeric characters, but cannot include blank spaces.

## Modes

Authentication configuration mode

## Usage Guidelines

The **no** form disables MAC authentication password override.

The MAC address is still the user name and cannot be changed.

## Examples

The following example enables MAC authentication password override on the device.

```
device(config)# authentication
device(config-authen)# mac-authentication password-override password
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# mac filter

Configures MAC address filters.

## Syntax

**mac filter** *filter-num* { **permit** | **deny** } { *source-mac source-mask* | **any** } { *destination-mac destination-mask* | **any** } [ **mirror** ]

**no mac filter** *filter-num* { **permit** | **deny** } { *source-mac source-mask* | **any** } { *destination-mac destination-mask* | **any** } [ **mirror** ]

## Command Default

MAC address filters are not configured.

## Parameters

*filter-num*
Configures the MAC address filter ID. You can configure up to 507 MAC address filters. The default value is 512.

**permit**
Permits the traffic.

**deny**
Denies the traffic.

*source-mac*
Configures the source Ethernet MAC address.

*source-mask*
Specifies the mask using f (ones) and zeros.

**any**
Configures the filter to match all source MAC addresses.

*destination-mac*
Configures the destination Ethernet MAC address.

*destination-mask*
Specifies the mask using f (ones) and zeros.

**any**
Configures the filter to match all destination MAC addresses.

**mirror**
Mirrors traffic that matches against configured entry.

## Modes

Global configuration mode

# Usage Guidelines

Once the MAC address filters are configured, you must apply the MAC address filters to a port.

The **no** form of the command removes the MAC address filters.

# Examples

The following example shows how to configure and apply MAC address filters. In this example, filter 1 is configured to deny traffic with a source MAC address that begins with "3565" to any destination, and filters 2 through 5 are configured to deny traffic with the specified destination MAC addresses. Filter 1024 permits all traffic that is not denied by any other filter.

```
device(config)# mac filter 1 deny 0000.0075.3676 ffff.0000.0000
device(config)# mac filter 2 deny any ffff.ffff.ffff ffff.ffff.ffff
device(config)# mac filter 3 deny any 0180.c200.0000 ffff.ffff.fff0
device(config)# mac filter 4 deny any 0000.0034.5678 ffff.ffff.ffff
device(config)# mac filter 5 deny any 0000.0045.6789 ffff.ffff.ffff
device(config)# mac filter 1024 permit any any
```

# mac filter enable-accounting

Enables access control list (ACL) accounting on Layer 2 MAC filters.

## Syntax

**mac filter** *num* **enable-accounting**

**no mac filter** *num* **enable-accounting**

## Command Default

This option is disabled.

## Parameters

*num*

Specifies the MAC filter ID.

**enable-accounting**

Enables MAC filter accounting on the specified interface.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command disables ACL accounting on the associated Layer 2 MAC filter interface.

## Examples

The following example enables ACL accounting on a Layer 2 MAC filter.

```
device(config)# mac filter 1 permit 0000.0000.0001 ffff.ffff.ffff any
device(config)# mac filter 1 enable-accounting
device(config)# interface ethernet 1/3/21
device(config-if-e1000-1/3/21)# mac filter-group 1
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# mac filter log-enable

Globally enables logging for MAC address filtered packets.

## Syntax

**mac filter log-enable**

**no mac filter log-enable**

## Command Default

Logging is disabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables logging of MAC address filtered packets.

## Examples

The following example globally enables logging for MAC address filtered packets.

```
device(config)# mac filter log-enable
```

# mac filter-group

Applies a group of MAC address filters to a port.

## Syntax

**mac filter-group** *filter-num* [ [ *filter-num* **to** *filter-num* | *filter-num* ] ...]

**no mac filter-group** *filter-num* [ [ *filter-num* **to** *filter-num* | *filter-num* ] ...]

## Command Default

MAC address filters are not applied to any port.

## Parameters

*filter-num*
Specifies the MAC address filter ID.

**to** *filter-num*
Specifies the range of MAC address filter IDs.

## Modes

Interface configuration mode

## Usage Guidelines

When applying the filter group to the interface, specify each line to be applied separately or use the **to** keyword to apply a consecutive range of filter lines, for example, 1 3 to 8 10.

The filters must be applied as a group. For example, if you want to apply four filters to an interface, they must all appear on the same command line.

You cannot add or remove individual filters in the group. To add or remove a filter on an interface, apply the filter group again containing all the filters you want to apply to the port. If you apply a filter group to a port that already has a filter group applied, the older filter group is replaced by the new filter group.

The **no** form of the command removes the MAC address filters configured on a port.

## Examples

The following exmaple configures MAC address filters and applies them to a port.

```
device(config)# mac filter 1 deny 0000.0075.3676 ffff.0000.0000
device(config)# mac filter 2 deny any ffff.ffff.ffff ffff.ffff.ffff
device(config)# mac filter 3 deny any 0180.c200.0000 ffff.ffff.fff0
device(config)# mac filter 4 deny any 0000.0034.5678 ffff.ffff.ffff
device(config)# mac filter 5 deny any 0000.0045.6789 ffff.ffff.ffff
device(config)# mac filter 1024 permit any any
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# mac filter-group 1 to 5 1024
```

# mac filter-group log-enable

Enables logging for MAC address filtered packets on a specific port.

## Syntax

**mac filter-group log-enable**

**no mac filter-group log-enable**

## Command Default

Logging for MAC address filtered packets on specific ports is disabled.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables logging for MAC address filtered packets on specific ports.

When a MAC address filter is applied to or removed from an interface, a syslog message is generated.

## Examples

The following example enables logging for filtered packets on the Ethernet interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# mac filter-group log-enable
```

# mac-learn-disable

Disables a physical port from automatic learning the source MAC address.

## Syntax

**mac-learn-disable**

**no mac-learn-disable**

## Command Default

By default, when a packet with an unknown source MAC address is received on a port, the device learns this MAC address on the port.

## Modes

Interface configuration mode

## Usage Guidelines

This command is not available on virtual routing interfaces. Also, if this command is configured on the LAG virtual interface, MAC address learning (source MAC address) will be disabled on all the ports in the LAG.

Entering the command on a tagged ports disables source MAC address learning for that port in all VLANs of which that port is a member. For example, if tagged port 1/1/1 is a member of VLAN 10, 20, and 30 and you issue the **mac-learn-disable** command on port 1/1/1, port 1/1/1 will not learn source MAC addresses, even if it is a member of VLAN 10, 20, and 30.

The **no** form of the command allows a physical port to learn source MAC addresses.

## Examples

The following example disables the automatic learning of the source MAC address.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# mac-learn-disable
```

# mac-notification interval

Configures the MAC-notification interval between each set of generated traps.

## Syntax

**mac-notification interval** *secs*

**no mac-notification interval** *secs*

## Command Default

No interval for MAC-notification is configured.

## Parameters

*secs*

Specifies the MAC-notification interval in seconds between each set of traps that are generated. The range is from 1 through 3600 seconds (1 hour). The default interval is 3 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command sets the interval to its default value, which is 3 seconds.

A trap is sent aggregating the MAC events such as addition or deletion depending on the interval you specify.

## Examples

The following example configures an interval of 40 seconds.

```
device(config)# mac-notification interval 40
```

The following example sets the interval to its default value:

```
device(config)# no mac-notification interval 3
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# mac-movement notification

Enables movement notifications and collects statistics for the movement of MAC addresses.

## Syntax

**mac-movement notification** { **interval-history** *seconds* | **threshold-rate** *moves* **sampling-interval** *seconds* }

**no mac-movement notification** { **interval-history** *seconds* | **threshold-rate** *moves* **sampling-interval** *seconds* }

## Parameters

**interval-history** *seconds*
: Configures the time interval during which the MAC address movement notification data is collected and enables a corresponding SNMP trap.

**threshold-rate** *moves*
: Configures the number of times a MAC address can move within the specified period until an SNMP trap is sent.

**sampling-interval** *seconds*
: Configures the sampling interval.

## Modes

Global configuration mode

## Usage Guidelines

The interval history includes statistical information such as the number of MAC addresses that move over the specified period, the total number of MAC address moves, which MAC addresses have moved, and how many times a MAC address has moved.

There is an upper limit on the number of MAC addresses for which MAC address-specific data is collected. This limit is necessary because it is not possible to report on all MAC addresses when many move.

Avoid threshold rates and sampling intervals that are too small. If you choose a small threshold and a sampling interval that is also small, an unnecessary high number of traps could occur.

The **no** form of the command disables movement notifications and stops collecting statistics for the movement of MAC addresses.

## Examples

The following example sets the notification interval to 300 seconds.

```
device(config)# mac-movement notification interval-history 300
```

The following example sets the notification for 500 moves and a sampling interval of 400 seconds.

```
device(config)# mac-movement notification threshold-rate 500 sampling-interval 400
```

# macsec cipher-suite

Enables GCM-AES-128 bit encryption or GCM-AES-256 bit integrity checks on MACsec frames transmitted between group members.

## Syntax

**macsec cipher-suite** { **gcm-aes-128** | **gcm-aes-128 integrity-only** | **gcm-aes-256** | **gcm-aes-256 integrity-only** }

**no macsec cipher-suite** { **gcm-aes-128** | **gcm-aes-128 integrity-only** | **gcm-aes-256** | **gcm-aes-256 integrity-only** }

## Command Default

GCM-AES encryption or integrity checking is not enabled. Frames are encrypted starting with the first byte of the data packet, and ICV checking is enabled.

## Parameters

**gcm-aes-128**
Enables GCM-AES-128 bit encryption.

**gcm-aes-128 integrity-only**
Enables GCM-AES-128 bit integrity checks.

**gcm-aes-256**
Enables GCM-AES-128 bit encryption.

**gcm-aes-256 integrity-only**
Enables GCM-AES-128 bit integrity checks.

## Modes

dot1x-mka-cfg-group mode

## Usage Guidelines

The **no** form of the command restores the default encryption and integrity checking.

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The **macsec cipher-suite** command can be used in conjunction with an encryption offset configured with the **macsec confidentiality-offset** command.

## Examples

The following example enables GCM-AES-128 encryption on group test1.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# macsec cipher-suite gcm-aes-128
```

The following example enables GCM-AES-128 bit integrity checking on test1.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# macsec cipher-suite gcm-aes-128 integrity-only
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on the ICX 7450 device. |
| 08.0.70 | ICX 7650 MACsec support was added. The command was modified to add GCM-AES-256 encryption options. |

# macsec confidentiality-offset

Configures the offset size for MACsec encryption.

## Syntax

**macsec confidentiality-offset** *size*

**no macsec confidentiality-offset** *size*

## Command Default

The default value for the MACsec encryption offset size is zero (0).

## Parameters

*size*

Determines where encryption begins. Valid values are:

- 30: Encryption begins at byte 31 of the data packet.
- 50: Encryption begins at byte 51 of the data packet.

## Modes

dot1x-mka-cfg-group mode

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The **no** form of the command disables encryption offset on all interfaces in the MACsec MKA group.

This command is only meaningful when encryption is enabled for the MACsec group using the **macsec cipher-suite** command.

## Examples

The following example configures a 30-byte offset on encrypted transmissions as part of group test1 parameters.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka)# macsec cipher-suite gcm-aes-128
device(config-dot1x-mka-group-test1)# macsec confidentiality-offset 30
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on the ICX 7450 device. |

| Release version | Command history |
|---|---|
| 08.0.70 | ICX 7650 MACsec support was added. The command was modified to add GCM-AES-256 encryption options. |

# macsec frame-validation

Enables validation checks for frames with MACsec headers and configures the validation mode (strict or not strict).

## Syntax

**macsec frame-validation** { **disable** | **check** | **strict** }

**no macsec frame-validation** { **disable** | **check** | **strict** }

## Command Default

MACsec frame validation is disabled (not visible in configuration).

## Parameters

**disable**

Disables validation checks for frames with MACsec headers.

**check**

Enables validation checks for frames with MACsec headers and configures non-strict validation mode. If frame validation fails, counters are incremented but packets are accepted.

**strict**

Enables validation checks for frames with MACsec headers and configures strict validation mode. If frame validation fails, counters are incremented and packets are dropped.

## Modes

dot1x-mka-cfg-group mode

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The **no** form of the restores the default (validation checks for frames with MACsec headers is disabled).

## Examples

The following example enables validation checks for frames with MACsec headers on group test1 and configures strict validation mode.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# macsec frame-validation strict
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on the ICX 7450 device. |
| 08.0.70 | ICX 7650 MACsec support was added. |

# macsec replay-protection

Specifies the action to be taken when packets are received out of order, based on their packet number. If replay protection is configured, you can specify the window size within which out-of-order packets are allowed.

## Syntax

**macsec replay-protection** { **strict** | **out-of-order** | **window-size** *size* } [ **disable** ]

**no macsec replay-protection** { **strict** | **out-of-order window-size** *size* } [ **disable** ]

## Command Default

## Parameters

**strict**

Does not allow out-of-order packets.

**out-of-order window-size**

Allows out-of-order packets within a specific window size.

*size*

Specifies the allowable window within which an out-of-order packet can be received. Allowable range is from 0 through 4294967295.

*disable*

Available only for the ICX 7450. Disables replay protection.

## Modes

dot1x-mka-cfg-group mode

## Usage Guidelines

This command is supported only on ICX 7450 and ICX 7650 devices.

The **no** form of the command disables macsec replay protection.

## Examples

The following example configures group test1 to accept packets in exact sequence only.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# macsec replay-protection strict
device(config-dot1x-mka-group-test1)#
```

The following example configures group test1 to accept out-of-order MACsec frames within a window size of 2000.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# macsec replay-protection out-of-order window-size 2000
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | The **disable** option for the **macsec replay-protection** command was introduced. Support for this command was added on ICX 7450 device. |
| 08.0.70 | ICX 7650 MACsec support was added. |

# management exclude

Excludes in-band and out-of-band (OOB) interfaces from management traffic.

## Syntax

**management exclude** { **all** | **http** | **ipv6ra** | **ntp** | **ssh** | **telnet** [ **inband** | **oob** ] }

**no management exclude**

## Command Default

None

## Parameters

**all**
Specifies ingress traffic for all applications. (Applicable to both switch and router images.)

**http**
Specifies requests for HTTP ingress connections. (Applicable to router images only.)

**ipv6ra**
Specifies IPv6 ingress Router Advertisement (RA) traffic. (Applicable to both switch and router images.)

**ntp**
Specifies NTP ingress traffic. (Applicable to router images only.)

**ssh**
Specifies requests for SSH ingress connections. (Applicable to router images only.)

**telnet**
Specifies requests for Telnet ingress connections. (Applicable to router images only.)

inband
Specifies in-band traffic only.

oob
Specifies OOB traffic only.

## Modes

Global configuration mode

## Usage Guidelines

The **management exclude** command is mutually exclusive with respect to either the **ip ssh strict-management-vrf** or the **telnet strict-management-vrf** commands. If the **management exclude** command is also configured, outbound SSH or Telnet connections are not blocked.

Use the **no** form of this command to remove all or one or more traffic types.

# Examples

To exclude OOB traffic for all applications:

```
device# configure terminal
device(config)# management exclude all oob
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# Related Commands

ip ssh strict-management-vrf, telnet strict-management-vrfip ssh strict-management-vrf, telnet strict-management-vrfip ssh strict-management-vrf, telnet strict-management-vrf

# management-vlan

Configures a VLAN to be a part of the management VLAN.

## Syntax

**management-vlan**

**no management-vlan**

## Command Default

VLAN configuration mode

## Modes

VLAN configuration mode

## Usage Guidelines

When this command is used, the out-of-band (OOB) interface port is not disabled. The port remains accessible to management even if in-band interface ports are busy forwarding packets at line rate. No packets are shared between the OOB management port and the in-band port.

The port is treated as an untagged port.

The **management-vlan** command is available only in the FastIron switch image.

Use the **no** form of the command to remove the VLAN from the management VLAN.

## Examples

To specify a VLAN and assign it to the management VLAN:

```
device# configure terminal
device(config)# vlan 20
device(config-vlan-20)# management-vlan
```

To remove the VLAN from the management VLAN:

```
device(config-vlan-20)# no management-vlan
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was introduced. |

# management-vrf

Configures a Virtual Routing and Forwarding (VRF) as a global management VRF.

## Syntax

**management-vrf** *vrf-name*

**no management-vrf** *vrf-name*

## Command Default

A management VRF is not configured.

## Parameters

*vrf-name*
Specifies the name of a preconfigured VRF.

## Modes

Global configuration mode

## Usage Guidelines

If the VRF is not preconfigured, command execution fails, and an error message is displayed. If you try to delete a management VRF that was not configured, the system displays an error message.

If a VRF is currently configured as the management VRF, it cannot be deleted or modified. Attempting to do so causes the system to return an error message. If a management VRF is already configured, you must remove the configuration before configuring a new one.

The **no** form of the command removes the management VRF. When the management VRF is deleted, a syslog message is displayed.

## Examples

The following example configures a management vrf.

```
device(config)# management-vrf mvrf
```

# map vlan (vxlan)

Maps VLAN to a VNI for a VXLAN overlay-gateway.

## Syntax

**map vlan** *vlan-id* **to VNI** *vni-id*

**no map vlan** *vlan-id* **to VNI** *vni-id*

## Command Default

VLAN to VNI is not mapped.

## Parameters

*vlan-id*

Identifies the VLAN to map to the VNI.

*vni-id*

Identifies the VXLAN Network Identifier (or VXLAN segment ID).

## Modes

Overlay-gateway configuration mode

## Usage Guidelines

The **no** form of the command removes the VLAN to VNI mapping.

The command is supported only on ICX 7750 devices.

A maximum of 256 VLAN-to-VNI mappings can be configured.

The designated VLAN must already have been configured.

The default VLAN, VLAN 1, unless the default is reconfigured, cannot be mapped to a VNI.

A VLAN with multicast snooping enabled cannot be mapped to a VNI.

A VLAN with a router interface cannot be mapped to a VNI.

A VLAN cannot be mapped to more than one VNI.

Multiple VLANs cannot be mapped to the same VNI.

## Examples

The following example maps VLAN 2 to VNI 3.

```
device# configure terminal
device(config)#overlay-gateway gate1
device(config-overlay-gate1)# type layer2-extension
device(config-overlay-gw-gate1)# map vlan 2 to vni 3
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# master

Configures the device as a Network Time Protocol (NTP) master clock to which peers synchronize themselves when an external NTP source is not available.

## Syntax

**master** [ **stratum** *number* ]

**no master** [ **stratum** *number* ]

## Command Default

The master clock is disabled by default.

## Parameters

**stratum** *number*

Specifies the NTP stratum number that the system will claim. The number can range from 2 to 15. The default value is 8.

## Modes

NTP configuration mode

## Usage Guidelines

Local time and time zone have to be configured before configuring the **master** command.

Use the **master** command with caution. It is very easy to override valid time sources using this command, especially if a low stratum number is configured. Configuring multiple machines in the same network with the **master** command can cause instability in timekeeping if the machines do not agree on the time.

**NOTE**
This command is not effective if NTP is enabled in client-only mode.

The **no** form of the command disables the master clock function.

## Examples

The following example configures the NTP master clock.

```
device(config)# ntp
device(config-ntp)# master stratum 5
```

# master (MRP)

Configures a node as the master node for the metro ring.

## Syntax

**master**

**no master**

## Command Default

A master node is not configured.

## Modes

MRP configuration mode

## Usage Guidelines

The **no** form of the command returns a master node a normal node.

Any node on a metro ring that does not have a shared interface can be designated as the ring master node. A master node can be the master node of more than one ring. However, if all nodes on the ring have shared interfaces, a node that does not have tunnel ports can be designated as the master node of that ring. If none of the nodes meet these criteria, you must change the priorities of the ring by reconfiguring the ring ID.

## Examples

The following example shows how to set a node as a master node.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-1)# name CustomerA
device(config-vlan-2-mrp-1)# master
device(config-vlan-2-mrp-1)# ring-interface ethernet 1/1/1 ethernet 1/1/2
device(config-vlan-2-mrp-1)# enable
```

# master-vlan

Adds the master VLAN to the topology group.

## Syntax

**master-vlan** *vlan-id*

**no master-vlan** *vlan-id*

## Command Default

A master VLAN is not configured.

## Parameters

*vlan-id*

Specifies the VLAN ID of the master VLAN.

## Modes

Topology group configuration mode

## Usage Guidelines

To configure a master VLAN, the VLAN must already be configured. Make sure all the Layer 2 protocol settings in the VLAN are correct for your configuration before you add the VLAN to the topology group. A topology group can have only one master VLAN. If you add a new master VLAN to a topology group that already has a master VLAN, the new master VLAN replaces the older master VLAN. All member VLANs and VLAN groups follow the Layer 2 protocol settings of the new master VLAN.

If you remove the master VLAN (by entering the **no master-vlan** command), the software selects the new master VLAN from member VLANs. A new candidate master VLAN is configured as a member VLAN so that the first added member VLAN will be a new candidate master VLAN. Once you save and reload, a member VLAN with the youngest VLAN ID will be the new candidate master. The new master VLAN inherits the Layer 2 protocol settings of the older master VLAN.

When removing the master VLAN from the topology group, Spanning Tree Protocol (STP) must be disabled on the master VLAN.

The **no** form of the command removes the master VLAN from the topology group.

## Examples

The following example adds the master VLAN 2 to the topology group 2.

```
device(config)# topology-group 2
device(config-topo-group-2)# master-vlan 2
```

# master-vlan (STP)

Adds the master VLAN to an STP group.

## Syntax

**master-vlan** *vlan-id*

**no master-vlan** *vlan-id*

## Command Default

The master VLAN is not configured.

## Parameters

*vlan-id*

Specifies the VLAN ID of the master VLAN.

## Modes

STP group configuration mode

## Usage Guidelines

To configure a master VLAN, the VLAN must already be configured. The master VLAN contains the STP settings for all the VLANs in the STP per VLAN group. An STP group can have only one master VLAN. If you add a new master VLAN to an STP group that already has a master VLAN, the new master VLAN replaces the older master VLAN.

If you remove the master VLAN (by entering the **no master-vlan** command), the software selects the new master VLAN from member VLANs. A new candidate master VLAN will be in configured as a member VLAN so that the first added member VLAN will be a new candidate master VLAN. Once you save and reload, a member VLAN with the youngest VLAN ID will be the new candidate master.

The **no** form of the command removes the master VLAN from the STP group.

## Examples

The following example adds the master VLAN 2 to the STP group 2.

```
device# configure terminal
device(config)# stp-group 2
device(config-stp-group-2)# master-vlan 2
```

# match address-local

Configures matching an Internet Key Exchange version 2 (IKEv2) policy based on a local IPv4 or IPv6 address.

## Syntax

**match address-local** { *ip-address ip-mask* | *ipv6-address ipv6mask* }

**no match address-local** { *ip-address mask* | *ipv6-address mask* }

## Command Default

The IKEv2 policy matches all local IPv4 or IPv6 addresses.

## Parameters

*ip-address ip-mask*
: Specifies a local IPv4 address and mask.

*ip-address ipv6-mask*
: Specifies a local IPv6 address and mask.

## Modes

IKEv2 policy configuration mode

## Usage Guidelines

The **no** form of the command restores the default value of the policy matching all local IPv4 or IPv6 addresses.

## Examples

The following example configures an IKEv2 policy named pol-mktg to use an IKEv2 proposal named prop-mktg and match the local IPv4 address 10.3.3.3 255.255.255.0.

```
device# configure terminal
device(config)# ikev2 policy pol-mktg
device(config-ike-policy-pol-mktg)# proposal prop-mktg
device(config-ike-policy-pol-mktg)# match address-local 10.3.3.3 255.255.255.0
device(config-ike-policy-pol-mktg# exit
```

The following example configures an IKEv2 policy named al2 to use an IKEv2 proposal and match the local IPv6 address 2001:100::1/64.

```
device# configure terminal
device(config)# ikev2 policy al2
device(config-ike-policy-al2)# proposal al2
device(config-ike-policy-al2)# match address-local 2001:100::1/64
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support was added for IPv6. |

# match as-path

Matches a BGP autonomous system path (AS-path) ACL in a route map instance.

## Syntax

**match as-path** *aspath-name* ...

**no match as-path** *aspath-name*

## Command Default

By default, match statements are not configured.

## Parameters

*aspath-name*
Specifies an AS-path access list.

## Modes

Route-map configuration mode

## Usage Guidelines

You can specify up to five AS-path ACLs.

The **no** form of the command removes the configuration.

## Examples

The following example configures a route map that matches based on AS-path “myas-path”.

```
device# configure termnial
device(config)# route-map myroutemap permit 1
device(config-routemap myroutemap)# match as-path myas-path
```

# match community

Matches a BGP community access list name in a route-map instance.

## Syntax

**match community** *name* [ **name ...** ] [ **exact-match** ]

**no match community** *name* [ **name ...** ] [ **exact-match** ]

## Command Default

By default, match statements are not configured.

## Parameters

*name*

Specifies a BGP community access list name.

*exact-match*

Specifies that an exact match is required.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of the command removes the configuration.

## Examples

The following example configures a route map that matches BGP community access list name “abccommunity”.

```
device# configure termnial
device(config)# route-map myroutemap permit 1
device(config-routemap myroutemap)# match community abccommunity
```

# match fvrf

Configures matching an Internet Key Exchange version 2 (IKEv2) policy based on a front-door virtual routing forwarding (fvrf) instance.

## Syntax

**match fvrf** { **vrf-name** *vrf* | **any** }

**no match fvrf** { **vrf-name** *vrf* | **any** }

## Command Default

An IKEv2 policy matches any VRF.

## Parameters

**vrf-name** *vrf*

Specifies matching a specific VRF.

**any**

Specifies matching any VRF.

## Modes

IKEv2 policy configuration mode

## Usage Guidelines

The **no** form of the command removes the specified forwarding VRF configuration.

## Examples

The following example shows how to create an IKEv2 policy named pol-mktg and configure it to use IKEv2 proposal prop-mktg and to match the policy based on a front-door VRF named mktg-vrf.

```
device(config)# ikev2 policy pol-mktg
device(config-ike-policy-pol-mktg)# proposal prop-mktg
device(config-ike-policy-pol-mktg)# match fvrf vrf-name mktg-vrf
device(config-ike-policy-pol-mktg# exit
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# match-identity

Configures match options for an Internet Key Exchange version 2 (IKEv2) profile based on local or remote identity parameters.

## Syntax

**match-identity local** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

**match-identity remote** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

**no match-identity local** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

**no match-identity remote** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

## Command Default

A match identity is not configured.

## Parameters

**local**
Specifies matching based on local identity.

**address** *ip-address*
Specifies matching based on a specific IPv4 address.

**address** *ipv6-address*
Specifies matching based on a specific IPv6 address.

**dn** *fqdn-name*
Specifies matching based on a specific Distinguished Name (DN).

**email** *email-address*
Specifies matching based on a specific email address.

**fqdn** *fqdn-name*
Specifies matching based on a specific fully qualified domain name (FQDN).

**key-id** *key-id*
Specifies matching based on a specific key ID.

**remote**
Specifies matching based on remote identity.

## Modes

IKEv2 profile configuration mode

# Usage Guidelines

An IKEv2 profile must contain an identity to match. When a match identity is not configured, the profile is considered incomplete and is not used. An IKEv2 profile can have more than one match identity. When multiple match statements of the same type are configured, a match occurs when any statement is matched.

The **no** form of the command removes the specified match identity configuration.

# Examples

The following example shows how to configure two match identities for an IKEv2 profile named prof-mktg, which is matched when the local IP address is 10.3.3.3. or the remote IP address is 10.2.2.1.

```
device# configure terminal
device(config)# ikev2 profile prof-mktg
device(config-ike-profile-prof-mktg)# match-identity local address 10.3.3.3
device(config-ike-profile-prof-mktg)# match-identity remote address 10.2.2.1
device(config-ike-profile-prof-mktg)# exit
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support was added for IPv6. |

# match interface

Configures the interface match clause in a route-map instance.

## Syntax

**match interface** { **ethernet** *stackid* **/***slot***/***port* | **loopback** *num* | **null0** | **tunnel** *number* | **ve** *vlan-id* } **...**

**no match interface** { **ethernet** *stackid* **/***slot***/***port* | **loopback** *num* | **null0** | **tunnel** *number* | **ve** *vlan-id* } **...**

## Parameters

**ethernet** *stackid slot port*
: Specifies an Ethernet interface with stackid, slot,and port numbers.

**loopback** *num*
: Specifies a loopback interface.

**null0**
: Specifies a loopback interface.

**tunnel** *num*
: Specifies a tunnel.

**ve** *vlan-id*
: Specifies a virtual Ethernet interface.

## Modes

Route-map configuration mode

## Usage Guidelines

A maximum of five interfaces is supported. There is no restriction on the number or type of each interface specified, as long as the total is less than or equal to five. The **no** form of the command removes the configuration.

## Examples

The following example configures a route map based on matching Ethernet interfaces 1/1/3 to 1/1/7.

```
device configure terminal
device(config)# route-map myroutemap permit 10
device(config-routemap myroutemap)# match interface ethernet 1/1/3 ethernet 1/1/7
```

# match ip address

Matches IP address conditions in a route map instance.

## Syntax

**match ip address** { *acl-name* | *acl-num* }

**match ip address prefix-list** *name*

**no match ip address** { *acl-name* | *acl-num* }

**no match ip address prefix-list** *name*

## Command Default

By default, match statements are not configured.

## Parameters

*acl-name*
Specifies an IPv4 ACL name.

*acl-num*
Specifies an IPv4 ACL number. Valid values range from 1 through 199.

**prefix-list** *name*
Specifies an IPv4 prefix list.

## Modes

Route-map configuration mode

## Usage Guidelines

You can specify up to five ACL names or ACL numbers. You can specify up to five IPv4 prefix lists

The **no** form of the command removes the configuration.

## Examples

The following example configures a route map that matches the standard ACL number 99.

```
device# configure terminal
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match ip address 99
```

The following example configures a route map that matches the IPv4 prefix list “myprefixlist”.

```
device# configure terminal
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match ip address prefix-list myprefixlist
```

# History

| Release version | Command history |
|---|---|
| 8.0.40a | Support was introduced for Ruckus ICX 7250 devices. |

# match ipv6 address

Matches IPv6 address conditions in a route map instance.

## Syntax

**match ipv6 address** *acl-name*

**match ipv6 address prefix-list** *prefix-list-name*

**no match ipv6 address** *acl-name*

**no match ipv6 address prefix-list** *prefix-list-name*

## Command Default

By default, match statements are not configured.

## Parameters

*acl-name*
Specifies an IPv6 ACL name.

**prefix-list** *prefix-list-name*
Specifies the name of an IPv6 prefix list.

## Modes

Route-map configuration mode

## Usage Guidelines

You can specify up to five ACL names. You can specify up to five IPv6 prefix lists

The **no** form of the command removes the **match ipv6 address** entry.

## Examples

The following example matches IPv6 routes that have addresses specified by the prefix list named “myprefixlist”.

```
device# configure terminal
device(config)# route-map extComRmap permit 10
device(config-route-map extComRmap)# match ipv6 address prefix-list myprefixlist
```

# match metric

Matches a route metric in a route-map instance.

## Syntax

**match metric** *value*

**no match metric** *value*

## Command Default

By default, match statements are not configured.

## Parameters

*value*

Matches a route metric for the route-map instance.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of the command removes the configuration.

## Examples

The following example configures a metric that matches on the specified value.

```
device# configure terminal
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match metric 1000
```

# match protocol

Matches the routes on protocol types and subtypes in a route-map instance.

## Syntax

**match interface** { **bgp** [ **external** | **internal** | **static-network** ] | **rip** | **static** }

**no match interface** { **bgp** [ **external** | **internal** | **static-network** ] | **rip** | **static** }

## Parameters

**bgp**
Matches BGP routes.

**external**
Matches eBGP routes.

**internal**
Matches iBGP routes.

**static-network**
Matches BGP static routes.

**rip**
Matches RIP routes.

**static**
Matches static routes.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of the command removes the configuration.

## Examples

The following example configures the RIP protocol as a matching criterion for a route-map instance.

```
device configure terminal
device(config)# route-map myroutemap permit 10
device(config-routemap myroutemap)# match protocol rip
```

# match route-type

Configures the route type clause in a route-map instance.

## Syntax

**match route-type** { **external-type1** | **external-type2** | **internal** }

**no match route-type** { **external-type1** | **external-type2** | **internal** }

## Parameters

**internal**
Specifies OSPF internal intra or inter type routes.

**external-type1**
Specifies OSPF external type 1 routes.

**external-type2**
Specifies OSPF external type 2 routes.

## Modes

Route-map configuration mode

## Usage Guidelines

The **no** form of the command removes the configuration.

## Examples

The following example configures OSPF external type 1 routes as a matching criterion for a route-map instance.

```
device# configure terminal
device(config)# route-map myroutemap permit 10
device(config-routemap myroutemap)# match route-type external-type1
```

# match tag

Matches a route tag in a route-map instance.

## Syntax

**match tag** *value*

**no match tag** *value*

## Parameters

*value*

Specifies a route tag and route tag value. Valid values range from 0 through 4294967294.

## Modes

Route-map configuration mode

## Usage Guidelines

A maximum of 8 tags can be configured.

The **no** form of the command removes the configuration.

## Examples

The following example specifies a route tag value of 6 as a matching criterion for a route-map instance.

```
device# configure terminal
device(config)# route-map myroutemap permit 10
device(config-routemap myroutemap)# match tag 6
```

# maxas-limit

Imposes a limit on the number of autonomous systems in the AS-PATH attribute.

## Syntax

**maxas-limit in** *num*

**no maxas-limit in**

## Parameters

**in**

Allows an AS-PATH attribute from any neighbor to impose a limit on the number of autonomous systems.

*num*

Specifies a value for the limit. Valid values range from 0 through 300. The default is 300.

## Modes

BGP configuration mode

## Examples

The following example sets the limit on the number of BGP4 autonomous systems in the AS-PATH attribute to 100.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# maxas-limit in 100
```

# maximum (port MAC security)

Configures the maximum number of secure MAC addresses an interface can store when port MAC security is enabled.

## Syntax

**maximum** *max-num*

**no maximum** *max-num*

## Command Default

By default, when port MAC security is enabled, an interface can store one secure MAC address.

## Parameters

*max-num*

The maximum number of secure MAC addresses that can be configured. The range is from 0 through 64, plus the total number of global resources available. The default is 1.

## Modes

Port security configuration mode

Port security interface configuration mode

## Usage Guidelines

Besides the maximum of 64 local resources available for an interface, 8192 additional global resources are shared among all interfaces on the device by default. The default system value of the global resources can be changed using the **system-max pms-global-pool** command. When an interface has secured enough MAC addresses to reach its configured limit for local resources, it uses the global resources to secure additional MAC addresses. Global resources are shared among all the interfaces on a first-come, first-served basis.

The **no** form of the command sets the maximum number of secure MAC addresses an interface can store to one.

## Examples

The following example configures the maximum number of secure MAC addresses an interface can store as 50.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port security
device(config-port-security-e1000-1/1/1)# maximum 50
```

# maximum-paths (BGP)

Sets the maximum number of BGP4 and BGP4+ shared paths.

## Syntax

**maximum-paths** *num* | **use-load-sharing**

**no maximum-paths**

## Parameters

*num*

Specifies the maximum number of paths across which the device balances traffic to a given BGP destination. Valid values range is from 1 through 32. The default is 1.

**use-load-sharing**

Uses the maximum IP ECMP path value that is configured by means of the **ip load-sharing** command.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use this command to change the maximum number of BGP4 shared paths, either by setting a value or using the value configured by the **ip load-sharing** command.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

The **no** form of the command restores the default.

## Examples

The following example sets the maximum number of BGP4 shared paths to 8.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# maximum-paths 8
```

The following example sets the maximum number of BGP4+ shared paths to that of the value already configured using the **ip load-sharing** command.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# maximum-paths use-load-sharing
```

The following example sets the maximum number of BGP4 shared paths to 2 in a nondefault VRF instance in the IPv4 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# maximum-paths 2
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# maximum-paths ebgp ibgp

Specifies the number of equal-cost multipath eBGP or iBGP routes or paths that are selected.

## Syntax

**maximum-paths** { **ebgp** *num* | **ibgp** *num* }

**no maximum-paths**

## Parameters

**ebgp**

Specifies eBGP routes or paths.

**ibgp**

Specifies iBGP routes or paths.

*num*

The number of equal-cost multipath routes or paths that are selected. Range is from 1 through 8. 1 disables equal-cost multipath.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

Enhancements to BGP load sharing support the load sharing of BGP4 and BGP4+ routes in IP Equal-Cost Multipath (ECMP), even if the BGP multipath load-sharing feature is not enabled by means of the **use-load-sharing** option to the **maximum-paths** command. You can set separate values for IGMP and ECMP load sharing. Use this command to specify the number of equal-cost multipath eBGP or iBGP routes or paths that are selected.

## Examples

The following example sets the number of equal-cost multipath eBGP routes or paths that will be selected to 6 in the IPv4 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)#  maximum-paths ebgp 6
```

The following example sets the number of equal-cost multipath iBGP routes or paths that will be selected to 4 in the IPv6 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# maximum-paths ibgp 4
```

The following example sets the number of equal-cost multipath eBGP routes or paths that will be selected to 3 in a nondefault VRF instance in the IPv4 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# maximum-paths ebgp 3
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# maximum-preference

Configures the Router Advertisement (RA) guard policy to accept RAs based on a router preference setting.

## Syntax

**maximum-preference** { **high** | **low** | **medium** }

**no maximum-preference** { **high** | **low** | **medium** }

## Command Default

The router preference setting for the RA guard policy is high (allows all RAs).

## Parameters

**high**

Configures the router preference of RAs for the RA guard policy to high (allows all RAs). This is the default.

**low**

Allows RAs of low router preference.

**medium**

Allows RAs of low and medium router preference.

## Modes

RA guard policy configuration mode

## Usage Guidelines

If a very low value is set, the RAs expected to be forwarded might get dropped.

The **no** form of this command removes the router preference for an RA guard policy.

## Examples

The following example configures the RA guard policy router preference to low:

```
device(config)# ipv6 raguard policy p1
device(config-ipv6-RAG-policy p1)# maximum-preference low
```

# max-hw-age

Enables and configures the maximum hardware age for denied MAC addresses.

## Syntax

**max-hw-age** *age*

**no max-hw-age** *age*

## Command Default

The default hardware aging time is 70 seconds.

## Parameters

*age*

Specifies the maximum hardware age in seconds. The possible values range from 1 to 65535 seconds.

## Modes

Authentication mode

## Usage Guidelines

Once the hardware aging period ends, the blocked MAC address ages out, and can be authenticated again if the device receives traffic from the MAC address.

The **no** form of this command disables maximum hardware age.

## Examples

The following example enables maximum hardware age and sets it to 160 seconds.

```
device(config)# authentication
device(config-authen)# max-hw-age 160
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# max-mcache

Configures the maximum number of PIM cache entries.

## Syntax

**max-mcache** *num*

**no max-mcache** *num*

## Command Default

If this command is not configured, the maximum value is determined by the **system max pim-hw-mcache** command or by available system resources.

## Parameters

*num*

Specifies the maximum number of multicast cache entries for PIM. Valid values range from 1 through 12288. The default is 12288.

## Modes

PIM router configuration mode

PIM router VRF mode

## Usage Guidelines

Configure the **max-mcache** command to define the maximum number of repeated cache entries for PIM traffic being sent from the same source address and being received by the same destination address. To define this maximum for the default VRF, configure the command in router PIM configuration mode; to define the maximum for a specific VRF, first configure the **router pim vrf** command.

The **no** form of the command restores the default.

## Examples

The following example configures the maximum number of PIM cache entries for the default VRF to 999.

```
device# configure terminal
device(config)# router pim
device(config-pim-router)# max-mcache 999
```

The following example configures the maximum number of PIM cache entries for VRF green to 888.

```
device# configure terminal
device(config)# router pim vrf green
device(config-pim-router-vrf-green)# max-mcache 88
```

# max-metric router-lsa (OSPFv2)

Advertises the maximum metric value in different Link State Advertisements (LSAs).

## Syntax

**max-metric router-lsa** [ **all-lsas** ] [ **all-vrfs** ] [ **external-lsa** *metric-value* ] [ **link** { **all** | **ptp** | **stub** | **transit** } ] [ **on-startup** { *time* | **wait-for-bgp** } ] [ **summary-lsa** *metric-value* ] [ **te-lsa** *metric-value* ]

**no max-metric router-lsa** [ **all-lsas** ] [ **all-vrfs** ] [ **external-lsa** *metric-value* ] [ **link** { **all** | **ptp** | **stub** | **transit** } ] [ **on-startup** { *time* | **wait-for-bgp** } ] [ **summary-lsa** *metric-value* ] [ **te-lsa** *metric-value* ]

## Parameters

**all-lsas**

Sets the **external-lsa**, **summary-lsa**, and **te-lsa** optional parameters to the corresponding default max-metric value.

**all-vrfs**

Applies the configuration change to all instances of OSPFv2.

**external-lsa** *metric-value*

Configures the maximum metric value for all external type-5 and type-7 LSAs. The range for metric value is 1 through 16777214 (0x00001 - 0x00FFFFFE), and the default is 16711680 (0x00FF0000).

**link**

Specifies the types of links for which the maximum metric is advertised. By default, the maximum metric is advertised only for transit links.

**all**

Advertises the maximum metric in Router LSAs for all supported link types.

**ptp**

Advertises the maximum metric in Router LSAs for point-to-point links.

**stub**

Advertises the maximum metric in Router LSAs for stub links.

**transit**

Advertises the maximum metric in Router LSAs for transit links. This is the default link type.

**on-startup**

Specifies the advertisement of the maximum metric for a limited period only, on startup.

*time*

Sets the time (in seconds) for which the specified links in Router LSAs are advertised when the metric is set to the maximum value of 0xFFFF. The range for *time* is 5 through 86400.

**wait-for-bgp**

Specifies that the maximum metric is advertised until BGP converges, or for 600 seconds.

**summary-lsa** *metric-value*

Configures the maximum metric value for all summary type 3 and type 4 LSAs. The range for metric value is 1 through 16777214 (0x00001 - 0x00FFFFFE), and the default is 16711680 (0x00FF0000).

**te-lsa** *metric-value*

Specifies that the TE metric field in the TE metric sub tlv for all type 10 Opaque LSAs LINK TLV originated by the router will be modified to the specified metric-value or a default value. The range for metric-value are 1 through 4294967295 (Hex: 0x00001 to 0xFFFFFFFF). The default value is 4294967295 (Hex: 0xFFFFFFFF). This parameter only applies to the default instance of OSPF.

# Modes

OSPFv2 router configuration mode

OSPFv2 router VRF configuration mode

# Usage Guidelines

Use this command to enable OSPFv2 to advertise its locally generated router LSAs with a maximum metric to direct transit traffic away from the device, while still routing for directly connected networks. By advertising the maximum metric, the device does not attract transit traffic.

Any new OSPFv2 instance configured after the max-metric configuration is completed requires that the **max-metric** command be configured again to take in the new OSPFv2 instance.

The **no** form of the command disables the advertising of the maximum metric value in different LSAs.

# Examples

The following example turns off the advertisement of special metric values in all router, summary, and external LSAs.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# no max-metric router-lsa
```

The following example configures an OSPFv2 device to advertise a maximum metric for 72 seconds after a restart before advertising with a normal metric.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# max-metric router-lsa on-startup 72
```

The following example indicates that OSPF is being shutdown and that all links in the router LSA should be advertised with the value 0xFFFF and the metric value for all external and summary LSAs is set to 0xFF0000 until OSPF is restarted. This configuration will not be saved.

```
device# configure terminal
device(config)# ip router ospf
device(config-ospf-router)# max-metric router-lsa external-lsa summary-lsa link all
```

# max-sessions

Specifies the maximum number of MAC sessions that can be authenticated per device or per port for MAC authentication and 802.1X authentication.

## Syntax

**max-sessions** *count*

**no max-sessions** *count*

## Command Default

The default number of MAC sessions that can be authenticated on a single interface is 2.

## Parameters

*count*

Specifies the maximum number of authenticated MAC sessions.

## Modes

Authentication configuration mode

## Usage Guidelines

The maximum number of authenticated MAC sessions on an interface depends on the ICX device and dynamic ACL assignments.

If RADIUS assigns dynamic ACL to at least one client on the interface, the maximum number of MAC sessions that can be authenticated is limited to 32 in all ICX devices.

If dynamic ACL is not assigned to any of the clients on the interface, the maximum number of MAC addresses that can be authenticated varies depending on the ICX device as specified in Table 5 on page 201.

System reload is not required for the changes to take effect. However, existing sessions are cleared for the changes to take effect.

**TABLE 8 Maximum number of authenticated MAC sessions per port on various platforms**

| Supported platforms | Maximum number of MAC sessions per port when none of the Clients has dynamic ACL | Maximum number of MAC sessions per port when at least one User has Dynamic ACL |
|---|---|---|
| ICX 7750 | 1024 | 32 |
| ICX 7650 | 1024 | 32 |
| ICX 7450 | 1024 | 32 |
| ICX 7250 | 1024 | 32 |

The system limit for authenticated MAC sessions also varies and depends on the ICX device and dynamic ACL assignments.

**TABLE 9 Maximum number of authenticated MAC sessions per system (standalone or stack) on various platforms**

| Supported platforms | Maximum number of MAC sessions per system when none of the clients has dynamic ACL | Maximum number of MAC sessions per system when at least one client has dynamic ACL |
|---|---|---|
| ICX 7750 | 1536 | 512 |
| ICX 7650 | 1536 | 512 |
| ICX 7450 | 1536 | 512 |
| ICX 7250 | 1536 | 512 |

The **no** form of the command reinstates the maximum authenticated MAC sessions allowed per port to the default value of 2.

# Examples

The following example specifies the maximum number of authenticated MAC sessions for the device.

```
device# configure terminal
device(config)# authentication
device(config-authen)# max-sessions 32
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# max-sw-age

Configures the maximum software aging.

## Syntax

**max-sw-age** *age*

**no max-sw** *age*

## Command Default

The default is 120 seconds.

## Parameters

*age*

You can specify from 1 - 65535 seconds.

## Modes

Authentication mode

## Usage Guidelines

Aging for a permitted or non-blocked MAC address occurs in two phases, known as MAC aging interval configured using the **mac-age-time** command and software aging. After normal MAC aging period for permitted clients (or clients in restricted VLAN), the software aging period begins. After the software aging period ends, the client session ages out and can be authenticated again if the device receives traffic from the MAC address.

Software aging is not applicable for blocked MAC addresses.

The **no** form of this command disables maximum software age.

## Examples

The following example configures the maximum software age to 170 seconds.

```
device(config)# authentication
device(config-authen)# max-sw-age 170
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# max-vlan (SPX)

Configures the maximum number of VLANs of which an 802.1br port extender (PE) port can be a member.

## Syntax

**max-vlan** *value*

**no max-vlan** *value*

## Command Default

By default, a port can be member of up to four VLANs.

## Parameters

*value*

Specifies the number of VLANs of which the port can be a member. The number of VLANs ranges from from 5 through 16.

## Modes

Interface configuration mode

## Usage Guidelines

**NOTE**
The max-vlan command is replaced beginning in FastIron release 08.0.80 by the **max-vlans-per-pe-port** command.

**NOTE**
This command is applicable to 802.1br PE virtual ports only. The command does not apply to physical ports.

The command allows you to add up to 128 PE ports to as many as 16 VLANs.

You can configure a higher or lower **max-vlan** value for the PE port. The new value must be greater than or equal to the current number of VLANs of which the port is a member.

If you try to add a PE port to more than the maximum number of allowed VLANs for that port, the system will throw an error such as “Error: maximum number of vlans allowed for PE port (x/y/z),vlans allowed (<max-vlan>) has been reached. Cannot add this port to vlan *xxx*."

The **no** form of the command restores the default number of VLANs.

## Examples

The following example configures the maximum number of VLANs of which a port can be a member to 12.

```
device(config)# interface ethernet 17/1/1
device(config-if-e1000-17/1/1)# max-vlan 12
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.80 | This command was deprecated. |

# max-vlans-per-pe-port (SPX)

Configures number of VLANs allowed per PE port in a Campus Fabric system.

## Syntax

**max-vlans-per-pe-port** { *number-allowed-vlans* }

**no max-vlans-per-pe-port**

## Command Default

By default, four VLANs are reserved per PE port, and the number of allowable VLANs per PE port is 32.

## Parameters

*number-allowed-vlans*
Number of VLANs allowed per PE port. The range of valid values is 5 through 1024 (decimal).

## Modes

CB configuration sub-mode

## Usage Guidelines

The **no** version of the command resets the allowable VLANs per PE port to 32.

The command replaces the **max-vlans** command in PE interface mode, which is deprecated in FastIron release 08.0.80.

This command is used to change the allowed number of VLANs per PE port from the default. Addition of PE ports into more than 4 VLANs is dependent on availability of PE port VLAN resources.

The global pool contains 4096 potential entries. Use the **show spx pe-port-vlan-resource** command to check how many entries are available.

The current non-default setting for **max-vlans-per-pe-port** is included in **show running-config** command output.

An error message is displayed in the following cases:

- You attempt to configure more than the maximum number of VLANs for a PE port.
- The PE port VLAN global pool is empty when you try to configure a VLAN for a PE port.
- A hash collision occurs when you attempt to configure a VLAN for a PE port.
- You have configured a **max-vlans-per-pe port** value and then try it set it to a value lower than the maximum VLANs that PE ports are currently configured as part of.

# Examples

The following example increases the maximum allowable VLANs per PE port to 64.

```
ICX7750-48F Router# configure terminal
ICX7750-48F Router(config)# spx cb-config
ICX7750-48F Router(config-spx-cb)# max-vlans-per-pe-port 64
ICX7750-48F Router(config-spx-cb)# end
ICX7750-48F Router#
```

The following example derives the **max-vlan-per-pe-port** value from show running-config output.

```
ICX7750-48F Router# show running-config | i max-vlans-per-pe-port
  max-vlans-per-pe-port 1024
ICX7750-48F Router#
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# med-missing-as-worst

Configures the device to favor a route that has a Multi-Exit Discriminator (MED) over a route that does not have one.

## Syntax

**med-missing-as-worst**

**no med-missing-as-worst**

## Modes

BGP configuration mode

## Usage Guidelines

When MEDs are compared, by default the device favors a low MED over a higher one. Because the device assigns a value of 0 to a route path MED if the MED value is missing, the default MED comparison results in the device favoring the route paths that do not have MEDs.

The **no** form of the command restores the default where a device does not favor a route that has a MED over other routes.

## Examples

The following example configures the device to favor a route containing a MED.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# med-missing-as-worst
```

# member-group

Adds the member VLAN group to the topology group.

## Syntax

**member-group** *number*

**no member-group** *number*

## Command Default

A member VLAN group is not added to the topology group.

## Parameters

*number*

Specifies the member VLAN group ID.

## Modes

Topology group configuration mode

## Usage Guidelines

The **no** form of the command removes the member VLAN group.

The VLAN group must already be configured.

Once you add a VLAN group as a member of a topology group, all the Layer 2 protocol configuration information for the VLAN group is deleted. For example, if STP is configured on a VLAN and you add the VLAN to a topology group, the STP configuration is removed from the VLAN. Once you add the VLAN to a topology group, the VLAN uses the Layer 2 protocol settings of the master VLAN. If you remove a member VLAN group from a topology group, you must reconfigure the Layer 2 protocol information in the VLAN group.

## Examples

The following example shows how to add a member VLAN group:

```
device(config)# topology-group 2
device(config-topo-group-2)# member-group 2
```

# member-group (STP)

Adds the member VLAN group to the STP group.

## Syntax

**member-group** *number*

**no member-group** *number*

## Command Default

A member VLAN group is not added to the STP group.

## Parameters

*number*

Specifies the member VLAN group ID.

## Modes

STP group configuration mode

## Usage Guidelines

The VLAN group must already be configured. All the VLANs in the member group inherit the STP settings of the master VLAN in the group.

The **no** form of the command removes the member VLAN group.

## Examples

The following example shows how to add a member VLAN group.

```
device(config)# stp-group 2
device(config-stp-group-2)# member-group 2
```

# member-vlan

Adds members to the VLAN topology group.

## Syntax

**member-vlan** *vlan-id* [ **to** *vlan-id* | [ *vlan-id* **to** *vlan-id* | *vlan-id*]... ]

**no member-vlan** *vlan-id* [ **to** *vlan-id* | [ *vlan-id* **to** *vlan-id* | *vlan-id*]... ]

## Command Default

Member VLANs are not added to the VLAN topology group.

## Parameters

*vlan-id*

Adds a member VLAN ID to the topology group.

**to** *vlan-id*

Adds the range of member VLANs to the topology group.

## Modes

Topology group configuration mode

## Usage Guidelines

The member VLAN group must configured before adding it to the topology group.

Each topology group can control up to 4096 VLANs. The VLANs within a VLAN group have the same ports and use the same values for other VLAN parameters.

Once you add a VLAN as a member of a topology group, all the Layer 2 protocol configuration information for the VLAN is deleted. For example, if STP is configured on a VLAN and you add the VLAN to a topology group, the STP configuration is removed from the VLAN. Once you add the VLAN to a topology group, the VLAN uses the Layer 2 protocol settings of the master VLAN. If you remove a member VLAN from a topology group, you must reconfigure the Layer 2 protocol information in the VLAN or VLAN group.

The **no** form of the command removes the member VLANs from the topology group.

## Examples

The following example adds the members to the VLAN topology group.

```
device(config)# topology-group 2
device(config-topo-group-2)# member-vlan 4
device(config-topo-group-2)# member-vlan 5
```

# member-vlan (STP)

Adds member VLANs to the STP group.

## Syntax

**member-vlan** *vlan-id* [ **to** *vlan-id* | [ *vlan-id* **to** *vlan-id* | *vlan-id*]... ]

**no member-vlan** *vlan-id* [ **to** *vlan-id* | [ *vlan-id* **to** *vlan-id* | *vlan-id*]... ]

## Command Default

Member VLANs are not added to the STP group.

## Parameters

*vlan-id*
: Adds a member VLAN ID to the STP group.

**to** *vlan-id*
: Adds the range of member VLANs to the STP group.

## Modes

STP group configuration mode

## Usage Guidelines

The member VLAN group must be configured before adding it to the STP group.

All the VLANs in the member group inherit the STP settings of the master VLAN in the group.

The **no** form of the command removes the member VLANs from the STP group.

## Examples

The following example adds the member VLANs to the STP group.

```
device(config)# stp-group 2
device(config-stp-group-2)# member-vlan 4
device(config-stp-group-2)# member-vlan 5
```

# mesh-group

Configures a multicast source discovery protocol (MSDP) mesh group from several rendezvous points (RPs).

## Syntax

**mesh-group** *group-name peer-address*

**no mesh-group** *group-name peer-address*

## Command Default

Mesh groups are not configured.

## Parameters

*group-name*

Specifies the mesh group as alphabetic characters. The limit is 31 characters.

*peer-address*

Specifies the IP address of the MSDP peer that is being placed in the mesh group. Each mesh group can include up to 32 peers.

## Modes

MSDP VRF configuration mode

## Usage Guidelines

The **no** form of this command removes mesh groups.

You must configure the **msdp-peer** command to configure the MSDP peers by assigning their IP addresses and the loopback interfaces before you configure a mesh group.

You can have up to four mesh groups in a multicast network. Each mesh group can include up to 15 peers.

Each device that will be part of a mesh group must have a mesh group definition for all the peers in the mesh-group.

## Examples

This example configures an MSDP mesh group on each device that will be included in the mesh group.

```
Device(config)# router msdp
Device(config-msdp-router)# msdp-peer 206.251.18.31 connect-source loopback 2
Device(config-msdp-router)# msdp-peer 206.251.19.31 connect-source loopback 2
Device(config-msdp-router)# msdp-peer 206.251.20.31 connect-source loopback 2
Device(config-msdp-router)# mesh-group GroupA 206.251.18.31
Device(config-msdp-router)# mesh-group GroupA 206.251.19.31
Device(config-msdp-router)# mesh-group GroupA 206.251.20.31
Device(config-msdp-router)# exit
```

# message-interval

Changes the default PIM Sparse join or prune message interval.

## Syntax

**message-interval** [ **vrf** *vrf-name* ] *interval*

**no message-interval** [ **vrf** *vrf-name* ] *interval*

## Parameters

**vrf** *vrf-name*
Specifies a VRF instance.

*interval*
Specifies the join or prune message interval in seconds. The range is 10 through 18724; the default is 60.

## Command Default

The join or prune interval is 60 seconds.

## Modes

PIM router configuration mode

PIM router VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the default; the join-prune interval is 60 seconds.

PIM Sparse join and prune messages inform other PIM Sparse routers about clients who want to become receivers (join) or stop being receivers (prune) for PIM Sparse groups.

**NOTE**
Configure the same join or prune message interval on all the PIM Sparse routers in the PIM Sparse domain. The performance of PIM Sparse can be adversely affected if the routers use different timer intervals.

## Examples

This example changes the PIM join or prune interval to 30 seconds.

```
Device(config)# ipv6 router pim
Device(config-ipv6-pim-router)# message-interval 30
```

This example changes the PIM join or prune interval on a VRF to 30 seconds.

```
Device(config)# ipv6 router pim vrf blue
Device(config-ipv6-pim-router-vrf-blue)# message-interval 30
```

# metric-type

Configures the default metric type for external routes.

## Syntax

**metric-type** { **type1** | **type2** }

**no metric-type** { **type1** | **type2** }

## Command Default

Type 2

## Parameters

**type1**

The metric of a neighbor is the cost between itself and the device plus the cost of using this device for routing to the rest of the world.

**type2**

The metric of a neighbor is the total cost from the redistributing device to the rest of the world.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default setting. You must specify a type parameter when using the **no** form.

## Examples

The following example sets the default metric type for external routes to type 1.

```
device# configure terminal
device(config)# router ospf
device(config-ospf6-router)# metric-type type1
```

The following example sets the default metric type for external routes to type 2.

# metro-ring

Adds a metro ring to a port-based VLAN and enters MRP configuration mode.

## Syntax

**metro-ring** *ring-id*

**no metro-ring** *ring-id*

## Command Default

A metro ring is not added to a port-based VLAN.

## Parameters

*ring-id*

Specifies the ID of the metro ring. The ring ID ranges from 1 through 1023. 256 is reserved for VSRP.

## Modes

VLAN configuration mode

## Usage Guidelines

If you plan to use a topology group to add VLANs to the ring, make sure you configure MRP on the topology group master VLAN.

If you want to add more than one metro ring to a port-based VLAN, use the **metro-rings** command.

The **no** form of the command removes the metro ring from the port-based VLAN.

## Examples

The following example shows how to add the metro ring to a port-based VLAN.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-1)#
```

# mdi-mdix

Enables or disables Media Dependent Interface (MDI) and Media Dependent Interface Crossover (MDIX) detection on all Gigabit Ethernet Copper ports.

## Syntax

**mdi-mdix** { **mdi** | **mdix** | **auto** }

**no mdi-mdix** [ **mdi** | **mdix** | **auto** ]

## Command Default

The auto MDI/MDIX detection feature is enabled on all Gigabit Ethernet copper ports.

## Parameters

**mdi**

Turns off automatic MDI/MDIX detection and defines a port as an MDI-only port.

**mdix**

Turns off automatic MDI/MDIX detection and defines a port as an MDIX-only port.

**auto**

Enables automatic MDI/MDIX detection on a port.

## Modes

Interface configuration mode

## Usage Guidelines

The auto MDI/MDIX detection feature can automatically correct errors in cable selection, making the distinction between a straight-through cable and a crossover cable insignificant. The command applies to copper ports only.

**NOTE**
The **mdi-mdix mdi** and **mdi-mdix mdix** commands work independently of auto-negotiation. Thus, these commands work whether auto-negotiation is turned on or off.

The **no** form of the command disables the specified mode.

## Examples

The following example turns off automatic MDI/MDIX detection and defines a port as an MDI-only port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# mdi-mdix mdi
```

The following example turns on automatic MDI/MDIX detection on a port that was previously set as an MDI or MDIX port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# mdi-mdix auto
```

# mirror-port

Configures port mirroring on individual ports.

## Syntax

**mirror-port ethernet** *stackid/slot/port* [ **input** | **output** ]

**no mirror-port ethernet** *stackid/slot/port* [ **input** | **output** ]

## Command Default

Ports are not mirrored.

## Parameters

**ethernet** *stackid/slot/port*
Specifies the Ethernet port to which mirrored traffic is copied.

**input**
Copies the ingress traffic.

**output**
Copies the egress traffic.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to configure ports to which the monitored traffic is copied. If you do not specify the traffic type, both types of traffic apply. The input and output mirroring ports can be on different ports.

All FastIron devices can have one mirroring port that monitors multiple ports, but cannot have multiple mirror ports for one monitored port. If the mirror port and the monitored ports are on different stack units, only one active mirror port is allowed for the entire traditional stack. If the mirror port and the monitored ports are on the same port region, multiple active mirror ports are allowed for the entire traditional stack. Devices in a traditional stack support 24 ports per port region.

**NOTE**
Port-based mirroring and VLAN-based mirroring cannot be enabled on a port at the same time.

The **no** form of the command removes the mirrored ports.

## Examples

The following example shows the port mirroring configuration.

```
device(config)# mirror-port ethernet 1/2/4
```

# mka-cfg-group

Creates and names a MACsec Key Agreement (MKA) configuration group.

## Syntax

**mka-cfg-group** *group-name*

**no mka-cfg-group** *group-name*

## Command Default

No MACsec options are configured for an MKA configuration group. All related parameters retain their default settings.

## Parameters

*group-name*

Provides a name for an MKA configuration group that can be applied to ports.

## Modes

dot1x-mka configuration mode

dot1x-mka-interface configuration mode

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The **no** form of this command deletes the MKA configuration group. MACSec is disabled on the ports where the group is configured.

The **dot1x-mka-enable** command must be executed before the **mka-cfg-group** command can be used.

After the MACsec Key Agreement (MKA) configuration group is created, you can apply the configured group and its settings to an interface being configured using the **mka-cfg-group** command in the dot1x-mka-interface configuration mode.

# Examples

The following example creates the MKA configuration group test1.

```
device(config)# dot1x-mka
  dot1x-mka-enable              Enable MACsec
device(config)# dot1x-mka-enable
device(config-dot1x-mka)#
device(config-dot1x-mka)# mka-cfg-group
  ASCII string   Name for this group
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)#

device(config-dot1x-mka-group-test1)# key-server-priority
  DECIMAL   Priority of the Key Server. Valid values should be between 0 and 255
device(config-dot1x-mka-group-test1)# key-server-priority 5
device(config-dot1x-mka-group-test1)#

device(config-dot1x-mka-group-test1)# macsec cipher-suite
  gcm-aes-128   GCM-AES-128 Cipher suite
device(config-dot1x-mka-group-test1)# macsec cipher-suite gcm-aes-128
device(config-dot1x-mka-group-test1)#

device(config-dot1x-mka-group-test1)# macsec confidentiality-offset
  30   Confidentiality offset of 30
  50   Confidentiality offset of 50
device(config-dot1x-mka-group-test1)# macsec confidentiality-offset 30
device(config-dot1x-mka-group-test1)#

device(config-dot1x-mka-group-test1)# macsec frame-validation
  check     Validate frames with secTAG and accept frames without secTAG
  disable   Disable frame validation
  strict    Validate frames with secTAG and discard frames without secTAG
device(config-dot1x-mka-group-test1)# macsec frame-validation strict
device(config-dot1x-mka-group-test1)#

device(config-dot1x-mka-group-test1)# macsec replay-protection
  out-of-order   Validate MACsec frames arrive in the given window size
  strict         Validate MACsec frames arrive in a sequence
device(config-dot1x-mka-group-test1)# macsec replay-protection strict
device(config-dot1x-mka-group-test1)#
```

The following example applies the previously configured MKA group test1 to ethernet interface 1/3/3.

```
device(config)# dot1x-mka-enable
device(config-dot1x-mka)# enable-mka ethernet 1/3/3
device(config-dot1x-mka-1/3/3)# mka-cfg-group test1
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.20a | This command was expanded to support the association of a configured MKA group and its settings to an interface at the interface configuration level. The **mka-group** command was deprecated as part of this change. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# module (SPX)

Manually configures SPX module information.

## Syntax

**module** *id module-name*

**no module** *id module-name*

## Command Default

SPX module information is learned and system-generated by default.

## Parameters

*id*

Identifies the module. Must be a number from 1 through 4.

*module-name*

Identifies the module type, for example, icx7450-24-port-management-module.

## Modes

SPX unit configuration mode (CB only)

## Usage Guidelines

The **no** form of the command followed by the module number and the exact module name removes the module from the SPX configuration.

When you create a reserved SPX unit, you must configure modules for the unit. The base module 1 must be configured before other modules.

When you configure a reserved SPX unit, the system will not generate default SPX ports or SPX LAGs for the unit.

The CB can add or remove a reserved module for a live PE unit.

## Examples

The following example configures module 1 for SPX unit 21.

```
device# configure terminal
device(config)# spx unit 21
device(config-spx-unit-21)# module 1 icx7450-48f-sf-port-management-module
device(config-spx-unit-21)# spx-port 21/2/4
device(config-spx-unit-21)# exit
device(config)# exit
```

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# monitor (ERSPAN)

Configures ERSPAN monitoring.

## Syntax

**monitor profile** *profile-number* { **both** | **input** | **output** }

**no monitor profile** *profile-number* { **both** | **input** | **output** }

## Command Default

Ports are not monitored.

## Parameters

**profile** *profile-number*
Specifies the ERSPAN profile to be used. The monitor port is specified in the profile.

**both**
Monitors both incoming and outgoing traffic on the monitor port.

**input**
Monitors the ingress traffic on the monitor port.

**output**
Monitors the egress traffic on the monitor port.

## Modes

Interface configuration mode

## Usage Guidelines

You must configure an ERSPAN profile before you can enable ERSPAN monitoring.

ERSPAN does not support VLAN monitoring.

The **no** form of the command disables ERSPAN monitoring on the port.

## Examples

The following example shows how to enable ERSPAN monitoring for ingress and egress traffic. The monitor port is 1/1/1, and the ERSPAN profile is 1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# monitor profile 1 both
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# monitor (LAG)

Monitors an individual port in a deployed LAG.

## Syntax

**monitor** { **ethe-port-monitored** *stackid/slot/port* | **named-port-monitored** *name* } [ **ethernet** *stackid/slot/port* ] { **input** | **output** | **both** }

**no monitor**{ **ethe-port-monitored** *stackid/slot/port* | **named-port-monitored** *name* } [ **ethernet** *stackid/slot/port* ] { **input** | **output** | **both** }

## Command Default

Traffic is not monitored on ports.

## Parameters

**ethe-port-monitored** *stackid/slot/port*
Specifies the Ethernet port to be monitored.

**named-port-monitored** *name*
Specifies the named port that you want to monitor.

**ethernet** *stackid/slot/port*
Specifies the mirror ports to be used and specifies the port to which the traffic analyzer is attached.

**input**
Monitors the incoming packets.

**output**
Monitors the outgoing packets.

**both**
Monitors both incoming and outgoing packets.

## Modes

LAG configuration mode

## Usage Guidelines

By default, when you monitor the LAG virtual interface, aggregated traffic for all the ports in the LAG is copied to the mirror port.

You can configure the device to monitor individual ports in a LAG including Ethernet ports or named ports. If a new port is added to a deployed LAG and if the entire LAG is monitored, the new port will also be mirrored by the same port monitoring traffic across the entire LAG.

**NOTE**
You can use only one mirror port for each monitored LAG port. You cannot configure mirroring on an undeployed LAG.

The **no** form of the command stops monitoring the traffic.

# Examples

The following is an example of monitoring traffic on an individual Ethernet port within a LAG.

```
device(config)# lag test2 dynamic id 1
device(config-lag-test2)# ports ethernet 1/1/1 ethernet 1/1/9
device(config-lag-test2)# monitor ethe-port-monitored 1/1/1 ethernet 1/1/9 input
```

The following example shows the monitoring of traffic on a named port.

```
device(config)# lag test2 dynamic id 2
device(config-lag-test2)# ports ethernet 1/1/1 ethernet 1/1/9
device(config-lag-test2)# monitor named-port-monitored port1 both
```

# monitor

Configures monitoring of the mirrored ports.

## Syntax

**monitor** [ **ethernet** *stackid/slot/port* ] { **both** | **input** | **output** }

**no monitor** [ **ethernet** *stackid/slot/port* ] { **both** | **input** | **output** }

## Command Default

Ports are not monitored.

## Parameters

**ethernet** *stackid/slot/port*
Specifies the mirror port to be used.

**both**
Monitors both incoming and outgoing traffic on the mirrored port.

**input**
Monitors the ingress traffic on the mirrored port.

**output**
Monitors the egress traffic on the mirrored port.

## Modes

Interface configuration mode

VLAN configuration mode

## Usage Guidelines

If you configure both ACL mirroring and ACL-based rate limiting on the same port, then all packets that match are mirrored, including the packets that exceed the rate limit. The same port cannot be both a monitored port and the mirror port. The same port can be monitored by one mirror port for ingress traffic and another mirror port for egress traffic. The mirror port cannot be a LAG port. More than one monitored port can be assigned to the same mirror port.

For stacked devices, if the ingress and egress analyzer ports are always network ports on the local device, each device may configure the ingress and egress analyzer port independently. However, if you need to mirror to a remote port, then only one ingress and one egress analyzer port are supported for the entire system.

The **no** form of the command stops monitoring the mirrored ports.

# Examples

The following example shows how to monitor the mirrored ports.

```
device(config)# interface ethernet 1/2/11
device(config-if-e1000-1/2/11)# monitor ethernet 1/2/4 both
```

The following example shows how to configure VLAN-based mirroring.

```
device(config)# mirror-port ethernet 1/1/21 input
device(config)# vlan 10
device(config-vlan-10)# monitor ethernet 1/1/21
device(config-vlan-10)# exit
device(config)# vlan 20
device(config-vlan-20)# monitor ethernet 1/1/21
device(config-vlan-20)# end
```

# monitor-profile

Configures a monitor port profile.

## Syntax

**monitor-profile** *profile-number* **type erspan**

**no monitor-profile** *profile-number*

## Command Default

ERSPAN is not configured.

## Parameters

*profile-number*

Specifies the profile number to configure. If the profile is new, assigns this number to the profile. Valid values are from 1 through 4.

**type erspan**

Specifies the type of profile. The only supported profile is **erspan**.

## Modes

Global configuration mode

## Usage Guidelines

The source IP can be any port on the router. The destination IP is the port on the destination host.

The **no** form of the command deletes the ERSPAN profile.

## Examples

The following example configures an ERSPAN profile. This profile sends mirrored traffic from a port on switch 2.2.2.2 to the host 1.1.1.1.

```
device(config)# monitor-profile 1 type erspan
device(config-monitor-profile 1)# source-ip 2.2.2.2
device(config-monitor-profile 1)# destination-ip 1.1.1.1
device(config-monitor-profile 1)# exit
```

The following example modifies the destination host in an ERSPAN profile.

```
device(config)# monitor-profile 1 type erspan
device(config-monitor-profile 1)# no destination-ip 1.1.1.1
device(config-monitor-profile 1)# destination-ip 3.3.3.3
device(config-monitor-profile 1)# exit
```

The following example deletes an ERSPAN profile.

```
device(config)# no monitor-profile 1
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# mount disk0

Mounts the filesystem of the external USB.

## Syntax

**mount disk0**

## Modes

User EXEC mode.

## Examples

This example mounts the filesystem of the external USB.

```
device# mount disk0
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# msdp-peer

Configures a multicast source discovery protocol (MSDP) peer.

## Syntax

**msdp-peer** *ip-address* [ **connect-source loopback** *num* | **shutdown** ]

**no msdp-peer** *ip-address* [ **connect-source loopback** *num* | **shutdown** ]

## Parameters

*ip-address*
Specifies the IP address of the MSDP peer.

**connect-source loopback**
Specifies the loopback interface you want to use as the source for sessions with the neighbor; it must be reachable within the VRF.

**shutdown**
Disables the MSDP peer. Configure this keyword at the MSDP router configuration mode level.

## Modes

MSDP router configuration mode

VRF configuration mode

## Usage Guidelines

The **no** form of the command with deletes the MSDP peer configuration. You should provide the IP address to identify the MSDP peer configuration that needs deletion. Use the **shutdown** option to disable the MSDP peer.

**NOTE**
The PIM Sparse rendezvous point (RP) is also an MSDP peer.

**NOTE**
Devices that run MSDP usually also run BGP. The source address used by the MSDP device is normally configured to be the same source address used by BGP.

It is strongly recommended that you specify the **connect-source loopback** keyword when iyou configure the **msdp-peer** command. If you do not, the device uses the IP address of the outgoing interface. You should also make sure that the IP address of the connect-source loopback is the source IP address used by the PIM RP and the BGP device.

## Examples

The following example configures a device with the address 205.216.162.1 as an MSDP peer.

```
device(config)# router msdp
device(config-msdp-router)# msdp-peer 205.216.162.1
```

The following example configures an MSDP peer on a VRF.

```
device(config)# router msdp
device(config-msdp-router)# msdp-peer 205.216.162.1
```

The following example adds an MSDP peer and specifies a loopback interface as the source interface for sessions with the peer. By default, the device uses the subnet address configured on the physical interface where you configure the peer as the source address for sessions with the peer.

```
device(config)# interface loopback 1
device(config-lbif-1)# ip address 9.9.9.9/32
device(config)# router msdp
device(config-msdp-router)# msdp-peer 2.2.2.99 connect-source loopback 1
```

# mstp admin-edge-port

Configures ports as operational edge ports.

## Syntax

**mstp admin-edge-port** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no mstp admin-edge-port** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

Ports are not configured as edge ports.

## Parameters

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Configures a specified Ethernet port as an edge port, or configures a range of ports as edge ports.

**lag** *lag-id* [ **to** *lag-id* ]

Configures the specified LAG or range of LAGs as edge ports.

## Modes

Global configuration mode

## Usage Guidelines

You can define specific ports as edge ports for the region in which they are configured to connect to devices (such as a host) that are not running STP, RSTP, or MSTP. If a port is connected to an end device such as a PC, the port can be configured as an edge port.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of the command removes a port from being an edge port.

## Examples

The following example shows how to configure an Ethernet port as an edge port.

```
device(config)# mstp admin-edge-port ethernet 1/3/1
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include the LAG ID option. |

# mstp admin-pt2pt-mac

Creates a point-to-point link between ports to increase the speed of convergence.

## Syntax

**mstp admin-pt2pt-mac** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no mstp admin-pt2pt-mac** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

By default, a point-to-point link is not available between ports.

## Parameters

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Configures the specified Ethernet port or port range as one end of a point-to-point link.

**lag** *lag-id* [ **to** *lag-id* ]

Specifies a LAG or a range of LAGs to serve as one end of a point-to-point link.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the point-to-point link on the ports.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

## Examples

The following example configures two Ethernet ports as endpoints for point-to-point links.

```
device(config)# mstp admin-pt2pt-mac ethernet 1/2/5 ethernet 1/4/5
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add LAG ID options. |

# mstp disable

Disables MSTP on interfaces.

## Syntax

**mstp disable** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no mstp disable** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

MSTP is not enabled by default.

## Parameters

**ethernet** [ **to** *unit/slot/port* ]
Disables MSTP on the specified Ethernet interface or range of interfaces.

**lag** *lag-id* [ **to** *lag-id* ]
Disables MSTP on a LAG virtual interface or on a range of LAG virtual interfaces.

## Modes

Global configuration mode

## Usage Guidelines

When a port is disabled for MSTP, the port blocks all the VLAN traffic that is controlled by Multiple Spanning Tree Protocol (MSTP) instance and the Common and Internal Spanning Tree (CIST) instances.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of the command enables MSTP.

## Examples

The following example shows how to disable MSTP.

```
device(config)# mstp disable ethernet 1/2/1
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# mstp edge-port-auto-detect

Automatically sets a port as an operational edge port.

## Syntax

**mstp edge-port-auto-detect**

**no mstp edge-port-auto-detect**

## Command Default

Ports are not automatically set as edge ports.

## Modes

Global configuration mode

## Usage Guidelines

You can configure a Layer 3 switch to automatically set a port as an operational edge port if the port does not receive any BPDUs from the time of link-up. If the port receives a BPDU later, the port is automatically reset to become an operational non-edge port.

**NOTE**
After configuring, it takes the port about three seconds longer to come to the enable state.

The **no** form of the command resets the port as a non-operational edge port.

## Examples

The following example shows how to automatically set ports as edge ports.

```
device(config)# mstp edge-port-auto-detect
```

# mstp force-migration-check

Triggers a port to force transmit an MSTP BPDU.

## Syntax

**mstp force-migration-check** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no mstp force-migration-check** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

Ports are not configured to force transmit MSTP BPDUs.

## Parameters

**ethernet** [ **to** *unit/slot/port* ]

Configures the specified Ethernet port or range of ports to force tranmist an MSTP BPDU.

**lag** *lag-id* [ **to** *lag-id* ]

Configures the specified LAG virtual interface or range of LAG virtual interfaces to force tranmist an MSTP BPDU.

## Modes

Global configuration mode

## Usage Guidelines

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of the command disables the force transmit of an MSTP BPDU.

## Examples

The following example triggers the port to transmit an MSTP BPDU.

```
device(config)# mstp force-migration-check ethernet 1/3/1
```

## History

| Release version | Command history |
|---|---|
| FI 08.0.61 | This command was modified to include LAG ID options. |

# mstp force-version

Configures the bridge to send BPDUs in a specific format.

## Syntax

**mstp force-version** *mode*

**no mstp force-version** *mode*

## Command Default

By default, the bridge sends the BPDUs in MSTP mode (3).

## Parameters

*mode*

Forces the bridge to send BPDUs in a specific format: 0 for STP compatibility mode, 2 for RSTP compatibility mode, and 3 for MSTP mode.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the mode to MSTP mode.

## Examples

The following example configures the bridge to forward BPDUs in STP compatibility mode.

```
device(config)# mstp force-version 0
```

# mstp forward-delay

Configures the length of time a port waits before it forwards an RST BPDU after a topology change.

## Syntax

**mstp forward-delay** *time*

**no mstp forward-delay** *time*

## Command Default

The default is 15 seconds.

## Parameters

*time*

Configures the time period a port waits before it forwards an RST BPDU after a topology change. The period ranges from 4 through 30 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the value to the default value of 15 seconds.

## Examples

The following example configures the time period the port waits before it forwards an RST BPDU after a topology change to 10 seconds.

```
device(config)# mstp forward-delay 10
```

# mstp hello-time

Configures the interval between two Hello packets.

## Syntax

**mstp hello-time** *time*

**no mstp hello-time** *time*

## Command Default

By default, the interval is 2 seconds.

## Parameters

*time*

The time interval between two Hello packets. The value ranges from 1 through 10 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the interval to the default (2 seconds).

## Examples

The following example configures the interval between two Hello packets to 5 seconds.

```
device(config)# mstp hello-time 5
```

# mstp instance

Configures a Multiple Spanning Tree Protocol (MSTP) instance that allows multiple VLANs to be managed by a single STP instance and supports per-VLAN STP. This allows you to use fewer spanning-tree instances to map to VLANs.

## Syntax

**mstp instance** *number* { **priority** *priority-num* | **vlan** *vlan-id* [ **to** *vlan-id* ] | **vlan-group** *group-id* | **lag** *lag-id* | **ethernet** *unit/slot/port* { **path-cost** *cost-value* [ **priority** *priority-value* ] | **priority** *priority-value* [ **path-cost** *cost-value* ] } }

**no mstp instance** *number* { **priority** *priority-num* | **vlan** *vlan-id* [ **to** *vlan-id* ] | **vlan-group** *group-id* | **lag** *lag-id* | **ethernet** *unit/slot/port* { **path-cost** *cost-value* [ **priority** *priority-value* ] | **priority** *priority-value* [ **path-cost** *cost-value* ] } }

## Command Default

No MSTP instances are configured. Any VLANs remain in the common, internal spanning tree (CIST) or are free.

## Parameters

*number*

Specifies the number for the instance of MSTP that you are configuring. You can specify up to 15 instances, identifying each, in MSTP mode, by a number in the range 1 through 4094. In MSTP mode, you cannot specify the value 0, which identifies the CIST. In MSTP+ mode, the range is 0 through 4094.

**priority** *priority-num*

Configures the priority for an MSTP instance. Valid values are from 0 through 61440 in increments of 4096. The default value is 32768.

**vlan** *vlan-id*

Assigns one or more VLANs or a range of VLANs to the MSTP instance.

**to** *vlan-id*

Assigns a range of VLANs to the MSTP instance.

**vlan-group** *group-id*

Assigns one or more VLAN groups to the MSTP instance.

**lag** *lag-id*

Configures LAG port parameters for the MSTP instance.

**ethernet** *unit/slot/port*

Configures Ethernet port parameters for the MSTP instance.

**path-cost** *cost-value*

Configures MSTP port path cost. Valid values are from 1 through 200000000.

**priority** *priority-value*

Specifies the forwarding preference for instances within a VLAN or on the device. You can specify a numeric value in the range 0 to 61440 in increments of 4096. A higher priority variable means a lower forwarding priority. The default value is 32768.

# Modes

Global configuration mode

# Usage Guidelines

The Ruckus implementation of MSTP allows you to assign VLANs or ranges of VLANs to an MSTP instance before or after they have been defined. If predefined, a VLAN will be placed in the MSTI that it was assigned to immediately when the VLAN is created. Otherwise, the default operation is to assign all new VLANs to the CIST. VLANs assigned to the CIST by default can be moved later to a specified MSTI.

The system does not allow an MSTI without any VLANs mapped to it. Consequently, removing all VLANs from an MSTI, deletes the MSTI from the system. The CIST by contrast will exist regardless of whether or not any VLANs are assigned to it. Consequently, if all VLANs are moved out of a CIST, the CIST will still exist and remain functional.

You can set a priority to the instance that gives it forwarding preference over lower priority instances within a VLAN or on the switch. A higher number for the priority variable means a lower forwarding priority.

The system does not allow an MSTP instance without any VLANs mapped to it; removing all VLANs from an MSTP instance deletes the instance from the system.

In MSTP+ mode, you can specify an instance number value of 0 because MSTP+ mode allows you to add VLANs to and remove VLANs from the CIST.

In MSTP mode, the **no** form of this command moves a VLAN or VLAN group from its assigned MSTP back into the CIST. In MSTP+ mode, the **no** form of this command assigns any VLAN as a free VLAN.

# Examples

The following example configures an MSTP instance and map VLANs 1 to 7 to it.

```
Device(config)# mstp instance 7 vlan 4 to 7
```

The following example specifies a priority of 8192 to MSTP instance 1.

```
Device(config)# mstp instance 1 priority 8192
```

# History

| Release version | Command history |
|---|---|
| FI 08.0.61 | This command was modified to include LAG ID options. |

# mstp max-age

Configures the amount of time the device waits to receive a Hello packet before it initiates a topology change.

## Syntax

**mstp max-age** *time*

**no mstp max-age** *time*

## Command Default

The default is 20 seconds.

## Parameters

*time*

The time period a device waits to receive a Hello packet before it initiates a topology change. The period ranges from 6 through 40 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the maximum age to the default value.

## Examples

The following example configures the maximum age to 20.

```
device(config)# mstp max-age 20
```

# mstp max-hops

Configures the maximum hop count.

## Syntax

**mstp max-hops** *count*

**no mstp max-hops** *count*

## Command Default

The default is 20 hops.

## Parameters

*count*

The maximum hop count. The number of hops ranges from 1 through 40.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the maximum hop count to the default value.

## Examples

The following example configures the maximum hop count to 20.

```
device(config)# mstp max-hops 20
```

# mstp name

Configures the MSTP name for the device.

## Syntax

**mstp name** *name*

**no mstp name** *name*

## Command Default

The default name for the device is blank (no name).

## Parameters

*name*

The MSTP name for the device.

## Modes

Global configuration mode

## Usage Guidelines

Each switch that is running MSTP should be configured with a name. The name applies to the switch that can have many different VLANs that can belong to many different MSTP regions.

The **no** form of the command resets the MSTP name to blank (no name).

## Examples

The following example configures the MSTP name as Device1.

```
device(config)# mstp name Device1
```

# mstp revision

Configures an MSTP revision number for the device.

## Syntax

**mstp revision** *number*

**no mstp revision***number*

## Command Default

The default MSTP revision number for a device is 0.

## Parameters

*number*

The revision level for MSTP. The MSTP revision number ranges from 0 through 65535.

## Modes

Global configuration mode

## Usage Guidelines

The MSRP revision number applies to the device that can have many different VLANs that can belong to many different MSTP regions.

The **no** form of the command sets the revision level to 0.

## Examples

The following example shows how to set the MSTP revision number for a device.

```
device(config)# mstp revision 4
```

# mstp root-protect timeout

Configures a root protection timeout value for MSTP root guard.

## Syntax

**mstp root-protect timeout** *value*

**no mstp root-protect timeout**

## Command Default

MSTP root guard is not enabled.

## Parameters

*value*

Timeout value in seconds. Range is 5 through 600. The default is 30.

## Modes

Interface configuration mode

## Usage Guidelines

Use the **no** form of this command to reset the timer to the default.

## Examples

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# mstp scope

Configures VLANs in Multiple Spanning Tee Protocol (MSTP) mode.

## Syntax

**mstp scope** { **all** | **pvst** }

**no mstp scope** { **all** | **pvst** }

## Command Default

No VLAN is under direct MSTP control.

## Parameters

**all**

Configures MSTP on all VLANs.

**pvst**

Configures MSTP in per-VLAN spanning tree (PVST) mode.

## Modes

Global configuration mode

## Usage Guidelines

MSTP is not operational until the **mstp start** command is configured. You cannot start MSTP+ unless at least one MSTP+ instance of MSTP+ is configured.

The **no** form of this command removes the MSTP PVST mode and restores the device to non-MSTP mode.

## Examples

The following example configures MSTP mode on all VLANs.

```
device(config)# mstp scope all
```

The following example enables MSTP in PVST mode.

```
device(config)# mstp scope pvst
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was modified to support the **pvst** keyword. |

# mstp start

Enables MSTP on the device.

## Syntax

**mstp start**

**no mstp start**

## Command Default

MSTP is disabled by default.

## Modes

Global configuration mode

## Usage Guidelines

MSTP scope must be enabled on the device before MSTP can be enabled.

The **no** form of the command disables MSTP on a device.

## Examples

The following example shows how to start MSTP on the device.

```
device(config)# mstp start
```

# mtu-exceed

Configures a port to forward traffic to a port with a smaller MTU size.

## Syntax

**mtu-exceed** { **forward** | **hard-drop** }

**no mtu-exceed** { **forward** | **hard-drop** }

## Command Default

Port does not forward traffic to a port with a smaller MTU size (hard-drop).

## Parameters

**forward**

Configures the port to fragment and forward a packet from a port with a larger MTU to a port with a smaller MTU.

**hard-drop**

Configures the port to resets to the default and removes the forward function.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command configures the port not to forward traffic to a port with smaller MTU size.

## Examples

The following example configures the port to fragment and forward a packet from a port with a larger MTU to a port with a smaller MTU.

```
device(config)# mtu-exceed forward
```

# multicast disable-igmp-snoop

Disables IGMP snooping for a specific VLAN when snooping is enabled globally.

## Syntax

**multicast disable-igmp-snoop**

**no multicast disable-igmp-snoop**

## Command Default

The global IGMP snooping setting applies.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command enables IGMP snooping on VLAN when IGMP snooping is enabled globally.

## Examples

The following example disables IGMP snooping on VLAN 20.

```
device(config)# vlan 20
device(config-vlan-20)# multicast disable-igmp-snoop
```

# multicast disable-pimsm-snoop

Disables PIM Sparse mode (SM) snooping for a specific VLAN when snooping is enabled globally.

## Syntax

**multicast disable-pimsm-snoop**

**no multicast disable-pimsm-snoop**

## Command Default

The global PIM SM snooping setting applies.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the global PIM SM snooping setting.

## Examples

This example disables PIM SM snooping on VLAN 20.

```
Device(config)#config vlan 20
Device(config-vlan-20)#multicast disable-pimsm-snoop
```

# multicast fast-convergence

Configures a device to listen to topology change events in Layer 2 protocols such as spanning tree, and then send general queries to shorten the convergence time.

## Syntax

**multicast fast-convergence**

**no multicast fast-convergence**

## Command Default

Fast convergence is not configured.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default; fast convergence is not configured.

If the Layer 2 protocol cannot detect a topology change, fast convergence may not work in some cases. For example, if the direct connection between two devices switches from one interface to another, the Rapid Spanning Tree protocol (802.1w) considers this optimization rather than a topology change. In this example, other devices do not receive topology change notifications, and cannot send queries to speed up the convergence. Fast convergence works well with the regular spanning tree protocol in this case.

## Examples

This example configures fast convergence on VLAN 70.

```
Device(config)#vlan 70
Device(config-vlan-70)#multicast fast-convergence
```

# multicast fast-leave-v2

Configures fast leave for IGMP V2.

## Syntax

**multicast fast-leave-v2**

**no multicast fast-leave-v2**

## Command Default

Fast leave for IGMP V2 is not configured.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default; fast leave for IGMP V2 is not configured.

When a device receives an IGMP V2 leave message, it sends out multiple group-specific queries. If no other client replies within the waiting period, the device stops forwarding traffic. When the **multicast fast-leave-v2** command is configured, and when the device receives a leave message, it immediately stops forwarding to that port. The device does not send group specific-queries. When the **multicast fast-leave-v2** command is configured on a VLAN, you must not have multiple clients on any port that is part of the VLAN.

In a scenario where two devices connect, the querier device should not be configured for fast-leave-v2 because the port might have multiple clients through the non-querier.

You can configure the **ip multicast leave-wait-time** command to set the number of queries and the waiting period.

## Examples

This example configures fast leave for IGMP on VLAN 10.

```
Device(config)#vlan 10
Device(config-vlan-10)#multicast fast-leave-v2
```

# multicast limit (enable)

Configures the maximum number of multicast packets allowed per second.

## Syntax

**multicast limit** *num* **kbps**

**no multicast limit** *num* **kbps**

## Command Default

Multicast rate limiting is disabled.

## Parameters

*num*

Specifies the maximum number of multicast packets per second. The value can be 1 to 8388607.

**kbps**

Enables byte-based limiting. The value can be 1 to Max Port Speed.

## Modes

Interface configuration mode

## Usage Guidelines

Use 0 or the **no** form of the command to disable limiting.

## Examples

The following example enables a multicast limit of 131072 kbps.

```
device(config)# interface ethernet 9/1/1
device(config-if-e1000-9/1/1)# multicast limit 131072 kbps
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | The command was introduced. |

# multicast limit (logging)

Enables Syslog logging of multicast packets.

## Syntax

**multicast limit** *num* **kbps** [ **log** ]

**no multicast limit** *num* **kbps** [ **log** ]

## Parameters

*num*

Specify a range of 1 through 8388607 packets per second or 1 for Max Port Speed.

**kbps**

Enables byte-based limiting. The value can be 1 to Max Port Speed.

**log**

Enables Syslog logging when the multicast limit exceeds *num* **kpbs** .

## Command Default

Multicast rate logging is disabled.

## Modes

Interface configuration mode

## Usage Guidelines

Use 0 or the **no** form of the command to disable limiting.

## Examples

The following example enables multicast logging when the configured limit exceeds 100 Kbps.

```
device(config)# interface ethernet 1/2/1
device(config-if-e1000-1/2/1)# multicast limit 100 kbps log
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | The command was introduced. |
| 8.0.40a | The command was modified to include the keyword **log**. |

# multicast pimsm-snooping

Enables PIM SM snooping for a specific VLAN.

## Syntax

**multicast pimsm-snooping** [ **prune-wait** ] *seconds*

**no multicast pimsm-snooping** [ **prune-wait** ] *seconds*

## Command Default

The global setting is applied.

## Parameters

**prune-wait** *seconds*

Specifies the amount of time a device waits after receiving a PIM prune message before removing the outgoing interface (OIF) from the forwarding entry. The range is 0 to 65535 seconds. The default is 5 seconds.

## Modes

VLAN configuration mode

## Usage Guidelines

The prune-wait time is necessary on a LAN where multiple receivers could be listening to the group; it gives them time to override the prune message. Configure the **multicast pimsm-snooping** command with the **prune-wait** keyword to modify the prune-wait time according to topology and PIM router configurations.

In accordance with RFC 4601, PIM routers delay pruning for 3.5 seconds by default, so configuring a lower prune-wait value may cause traffic disruption. You should configure a prune-wait value lower than 3.5 seconds only if the topology supports it, for example, if the group has only one receiver, and an immediate prune is needed.

The no form of the command disables PIM SM snooping for the VLAN. The **no** form of the command with the prune-wait keyword restores the default prune-wait time (5 seconds).

## Examples

The following example enables PIM SM snooping for VLAN 10.

```
device# configure termnial
device(config)# vlan 10
device(config-vlan-10)# multicast pimsm-snooping
```

The following example configures the prune-wait time to 7 seconds for VLAN 10.

```
device# configure termnial
device(config)# vlan 10
device(config-vlan-10)# multicast pimsm-snooping prune-wait 7
```

# History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was introduced. |

# multicast port-version

Configures the IGMP version on individual ports in a VLAN.

## Syntax

**multicast port-version** { **2** | **3** } { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no multicast port-version** { **2** | **3** } { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

The port uses the IGMP version configured globally or for the VLAN.

## Parameters

**2**
Configures IGMP version 2.

**3**
Configures IGMP version 3.

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]
Configures the designated version on the specified Ethernet port (or range of ports).

**lag** *lag-id* [ **to** *lag-id* ]
Configures the designated version on the specified LAG (or range of LAGs).

**to**
Specifies a range of ports or LAGs.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the IGMP version configured globally or for the VLAN.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

See the description of the **ip multicast version** command for information on how to configure the IGMP version globally.

See the description of the **multicast version** command for information on how to configure the IGMP version on a VLAN.

# Examples

This example configures ports 1/2/4, 1/2/5, and 1/2/6 to use IGMP version 3.

```
Device(config)# vlan 20
(config-vlan-20)# multicast port-version 3 ethernet 1/2/4 to 1/2/6
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# multicast proxy-off

Turns off proxy activity for static groups.

## Syntax

**multicast proxy-off**

**no multicast proxy-off**

## Command Default

Proxy activity is on.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default; proxy activity is on.

When a device is configured for static groups, it acts as a proxy and sends membership reports for the static groups when it receives general or group-specific queries. When a static group configuration is removed, the group is deleted from the active group table immediately. However, leave messages are not sent to the querier, and the querier must age out the group. You can configure the **multicast proxy-off** command to turn off proxy activity.

## Examples

This example turns off proxy activity for VLAN 20.

```
Device(config)#vlan 20
Device(config-vlan-20)#multicast proxy-off
```

# multicast querier-address

Configures the IPv4 querier address per VLAN.

## Syntax

**multicast querier-address** *A.B.C.D*

**no multicast querier-address** *A.B.C.D*

## Parameters

*A.B.C.D*

Specifies an IPv4 address as the multicast querier address.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command disables the IPv4 querier address functionality.

## Examples

The following example specifies an IPv4 address as the multicast querier address for the VLAN.

```
device# configure terminal
device(config)# vlan 100
device(config-vlan-100)# multicast querier-address 2.2.2.2
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# multicast6 querier-address

Configures the IPv6 querier address per VLAN.

## Syntax

**multicast6 querier-address** *X:X::X:X*

**no multicast6 querier-address** *X:X::X:X*

## Parameters

*X:X::X:X*

Specifies an IPv6 link local address as the multicast querier address.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command disables the IPv6 querier address functionality.

## Examples

The following example specifies an IPv6 address as the multicast querier address for the VLAN.

```
device# configure terminal
device(config)# vlan 100
device(config-vlan-100)# multicast6 querier-address FE80::44
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# multicast router-port

Configures a static router Ethernet port or LAG to receive multicast control and data packets.

## Syntax

**multicast router-port** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no multicast router-port** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

The device forwards all multicast control and data packets only to router ports that receive queries.

## Parameters

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Specifies the Ethernet port (or range of ports) you want to force traffic to.

**lag** *lag-id* [ **to** *lag-id* ]

Specifies the LAG (or range of LAGs) you want to force traffic to.

**to**

Specifies a range of ports or LAGs.

## Modes

VLAN configuration mode

## Usage Guidelines

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of this command restores the default, that is, the device forwards all multicast control and data packets only to router ports that receive queries.

## Examples

This example configures a static port on Ethernet 1/1/3 on VLAN 70.

```
device# configure terminal
device(config)# vlan 70
device(config-vlan-70)# multicast router-port ethernet 1/1/3
```

This example configures a list of static ports on VLAN 70.

```
device# configure terminal
device(config)# vlan 70
device(config-vlan-70)# multicast router-port ethernet 1/1/24 ethernet 1/6/24 ethernet 1/8/17
```

This example configures a range of static ports on VLAN 70.

```
device# configure terminal
device(config)# vlan 70
device(config-vlan-70)# multicast router-port ethernet 1/1/1 to 1/1/8
```

This example configures a combined range and list of static ports on VLAN 70.

```
device# configure terminal
device(config)# vlan 70
device(config-vlan-70)# multicast router-port ethernet 1/1/1 to 1/1/8 ethernet 1/1/24 ethernet 1/6/24
ethernet 1/8/17
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# multicast static-group

Configures a static IGMP group for a VLAN.

## Syntax

**multicast static-group** *ipv4-address* [ **count** *num* ] { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no multicast static-group** *ipv4-address* [ **count** *num* ] { **lag** *lag-id* [ **to** *lag-id* ] | **ethernet** *unit***/***slot***/***port* [ **ethernet** *unit***/***slot***/***port* | **to** *unit***/***slot***/***port* ] | **management** { *unit/slot/port* | *management-id* } }

## Command Default

The VLAN cannot forward multicast traffic to ports that do not receive IGMP membership reports.

## Parameters

*ipv4-address*
Specifies the address of the static group.

**count** *num*
Specifies a contiguous range of groups.

**ethernet** *unit***/***slot***/***port* [ **to** *unit***/***slot***/***port* ]
Specifies a port or a range of ports to be included in the group.

**LAG** *lag-id* [ **to** *lag-id* ]
Specifies the LAG or range of LAGs to be included in the group.

## Modes

VLAN configuration mode

## Usage Guidelines

A snooping-enabled VLAN cannot forward multicast traffic to ports that do not receive IGMP membership reports. You can configure the **multicast static-group** command to create a static group that applies to specific ports, allowing packets to be forwarded to them even though they have no client membership reports.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of this command removes the static group from the VLAN.

# Examples

This example configures a static group on VLAN 20 that contains ports 1/1/3 and 1/1/5 to 1/1/7.

```
device# configure terminal
device(config)# vlan 20
device(config-vlan-20)# multicast static-group 224.1.1.1 count 2 ethernet 1/1/3 ethernet 1/1/5 to 1/1/7
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# multicast tracking

Enables tracking and fast leave on VLANs.

## Syntax

**multicast tracking**

**no multicast tracking**

## Command Default

Tracking and fast leave are disabled.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default, that is, tracking and fast leave are disabled.

The membership tracking and fast leave features are supported for IGMP V3 only. If any port or any client is not configured for IGMP V3, the multicast tracking command is ignored.

## Examples

This example enables tracking and fast leave on VLAN 20.

```
Device(config)#vlan 20
Device(config-vlan-20)#multicast tracking
```

# multicast version

Configures the IGMP version for snooping on a VLAN.

## Syntax

**multicast version** [ **2** | **3** ]

**no multicast version**

## Command Default

The globally-configured IGMP version is used.

## Parameters

**2**

Configures IGMP version 2.

**3**

Configures IGMP version 3.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the globally configured version.

If an IGMP version is configured for an individual port, that port uses the version configured for it, not the VLAN version.

See the description of the **ip multicast version** command for information on how to configure the IGMP version globally.

See the description of the **multicast port-version** command for information on how to configure the IGMP version on an individual port

## Examples

This example configures IGMP version 3 on VLAN 20.

```
Device(config)#vlan 20
Device(config-vlan-20)#multicast version 3
```

# multicast6

Configures the multicast listening discovery (MLD) mode on the device to active or passive.

## Syntax

**multicast6** [ **active** | **passive** ]

**no multicast6** [ **active** | **passive** ]

## Command Default

MLD mode is passive.

## Parameters

**active**

Configures MLD active mode, that is, the device actively sends out MLD queries to identify IPv6 multicast groups on the network, and makes entries in the MLD table based on the group membership reports it receives from the network.

**passive**

Configures MLD passive mode, that is, the device forwards reports to router ports that receive queries. MLD snooping in the passive mode does not send queries. However, it does forward queries to the entire VLAN.

## Modes

VLAN configuration mode

## Usage Guidelines

The MLD mode configured on a VLAN overrides the mode configured globally.

The **no** form of this command returns the device to the previous MLD mode.

## Examples

The following example configures MLD mode as active on VLAN 20.

```
device(config)# vlan 20
device(config-vlan-20)# multicast6 active
```

# multicast6 disable-mld-snoop

Disables multicast listening discovery (MLD) snooping for a specific VLAN when snooping is enabled globally.

## Syntax

**multicast6 disable-multicast-snoop**

**no multicast6 disable-multicast-snoop**

## Command Default

The global MLD snooping setting applies.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the global MLD snooping setting.

## Examples

This example disables MLD snooping on VLAN 20.

```
Device(config)#vlan 20
Device(config-vlan-20)#multicast6 disable-multicast-snoop
```

# multicast6 disable-pimsm-snoop

When PIM6 SM snooping is enabled globally, overrides the global setting and disables it for a specific VLAN.

## Syntax

**multicast6 disable-pimsm-snoop**

**no multicast6 disable-pimsm-snoop**

## Command Default

The globally configured PIM6 SM snooping applies.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the globally configured PIM6 SM snooping.

The device must be in multicast listening discovery (MLD) passive mode before PIM6 SM snooping can be disabled.

## Examples

This example enables PIM6 SM traffic snooping on VLAN 20.

```
Device(config)# vlan 20
Device(config-vlan-20)#multicast6 disable-pimsm-snoop
```

# multicast6 fast-convergence

Configures a device to listen to topology change events in Layer 2 protocols such as spanning tree, and then send general queries to shorten the convergence time.

## Syntax

**multicast6 fast-convergence**

**no multicast6 fast-convergence**

## Command Default

Fast convergence is not configured.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default; fast convergence is not configured.

Configure the **multicast6 fast-convergence** command to allow a device to listen to topology change events in Layer 2 protocols, such as Spanning Tree, and send general queries to shorten the convergence time.

If the Layer 2 protocol cannot detect a topology change, fast convergence may not work in some cases. For example, if the direct connection between two devices switches from one interface to another, the Rapid Spanning Tree protocol (802.1w) considers this to be optimization rather than a topology change. In this case, other devices do not receive topology change notifications and cannot send queries to speed up convergence. The original spanning tree protocol does not recognize optimization actions, and fast convergence works in all cases.

## Examples

This example configures fast convergence on VLAN 70.

```
device# configure terminal
device(config)# vlan 70
device(config-vlan-70)# multicast6 fast-convergence
```

# multicast6 fast-leave-v1

Configures fast leave for multicast listening discovery Version 1 (MLDv1)

## Syntax

**multicast6 fast-leave-v1**

**no multicast6 fast-leave-v1**

## Command Default

The device forwards traffic to a port immediately upon receiving a leave message.

## Modes

VLAN configuration mode

## Usage Guidelines

When the **multicast6 fast-leave-v1** command is configured on a VLAN, make sure you do not have multiple clients on any port that is part of the VLAN. When two devices connect, the querier device should not have the **multicast6 fast-leave-v1** command configured because the port might have multiple clients through the non-querier.

You can configure the **ipv6 multicast leave-wait-time** command to configure the number of queries and the waiting period in seconds.

The **no** form of this command restores the device to forward traffic to a port immediately upon receiving a leave message. The device sends group-specific queries.

## Examples

The following example configures fast leave for MLDv1 on VLAN 20.

```
device(config)# vlan 20
device(config-vlan-20)# multicast6 fast-leave-v1
```

# multicast6 pimsm-snooping

Enables PIM6 SM traffic snooping on a VLAN.

## Syntax

**multicast6 pimsm-snooping** [ **prune-wait** *time* ]

**no multicast6 pimsm-snooping** [ **prune-wait** *time* ]

## Command Default

PIM6 SM traffic snooping is disabled.

## Parameters

**prune-wait** *time*

Configures the amount of time a device waits after receiving a PIM prune message before removing the outgoing interface (OIF) from the forwarding entry. The value can be 0 to 30 seconds. The default is 5 seconds.

## Modes

VLAN configuration mode

## Usage Guidelines

The device must be in multicast listening discovery (MLD) passive mode before it can be configured for PIM6 sparse mode (SM) snooping.

When PIM6 SM snooping is enabled globally, you can override the global setting and disable it for a specific VLAN. You must configure the **multicast6 disable-pimsm-snoop** command to disable PIM6 SM snooping on a VLAN.

A smaller prune wait value reduces flooding of unwanted traffic. A prune wait value of zero causes the PIM device to stop traffic immediately upon receiving a prune message. If there are two or more neighbors on the physical port, then the **prune-wait** option should not be used because one neighbor may send a prune message while the other sends a join message at the same time, or within less than five seconds.

The **no** form of this command without options disables PIM6 SM traffic snooping on a VLAN. The **no** form of the command with the **prune-wait** option resets the prune-wait time as 5 seconds.

## Examples

The following example first configures VLAN 20 and adds the ports that are connected to the device and host in the same port-based VLAN . Then it enables MLD snooping passive on VLAN 20 and enables PIM6 SM traffic snooping on it. The prune-wait timer is set as 10 seconds.

```
device(config)# vlan 20
device(config-vlan-20)# untagged ethernet 1/1/5 ethernet 1/1/7 ethernet 1/1/11
device(config-vlan-20)# multicast6 passive
device(config-vlan-20)# multicast6 pimsm-snooping
device(config-vlan-20)# multicast6 pimsm-snooping prune-wait 10
```

# multicast6 port-version

Configures the multicast listening discovery (MLD) version on individual ports in a VLAN.

## Syntax

**multicast6 port-version** { **1** | **2** } { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no multicast6 port-version** { **1** | **2** } { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

The port uses the MLD version configured globally or for the VLAN.

## Parameters

**1**

Configures MLD version 1.

**2**

Configures MLD version 2.

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Specifies the port (or range of ports) to configure the version on. You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

**lag** *lag-id* [ **to** *lag-id* ]

Configures the designated version on the specified LAG (or range of LAGs).

**to**

Specifies a range of ports or LANs.

## Modes

VLAN configuration mode

## Usage Guidelines

When you configure the MLD version on a specified port or range of ports, the other ports use the MLD version specified with the **multicast6 version** command, or the globally configured MLD version.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of this command restores the MLD version configured globally or for the VLAN.

# Examples

This example configures ports 1/1/4, 1/1/5, 1/1/6, and 1/2/1 on VLAN 20 to use MLD version 2.

```
Device(config)# vlan 20
Device(config-vlan-20)# multicast6 port-version 2 ethernet 1/2/1 ethernet 1/1/4 to 1/1/6
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# multicast6 proxy-off

Turns off multicast listening discovery (MLD) proxy activity.

## Syntax

**multicast6 proxy-off**

**no multicast6 proxy-off**

## Command Default

MLD snooping proxy activity is on.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default; proxy activity is on.

When a device is configured for static groups, it acts as a proxy and sends membership reports for the static groups when it receives general or group-specific queries. When a static group configuration is removed, the group is deleted from the active group table immediately. However, leave messages are not sent to the querier, and the querier must age out the group. You can configure the **multicast proxy-off** command to turn off proxy activity.

## Examples

This example turns off proxy activity for VLAN 20.

```
Device(config)#vlan 20
Device(config-vlan-20)#multicast6 proxy-off
```

# multicast6 router-port

Configures a static router port to receive IPv6 multicast control and data packets.

## Syntax

**multicast6 router-port** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no multicast6 router-port** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

The device forwards all IPv6 multicast control and data packets only to router ports that receive queries.

## Parameters

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Specifies the Ethernet port, port list, or range of ports you want to force traffic to.

**lag** *lag-id* [ **to** *lag-id* ]

Specifies the LAG, set of LAGs, or range of LAGs you want to force traffic to.

**to**

Specifies a range of ports or LAGs.

## Modes

VLAN configuration mode

## Usage Guidelines

All multicast control and data packets are forwarded to router ports that receive queries. Although router ports are learned, you can configure static router ports to force multicast traffic to specific ports, even though these ports never receive queries.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of this command restores the default, that is, the device forwards all multicast control and data packets only to router ports that receive queries.

## Examples

This example configures a range and a list of static ports on VLAN 70.

```
device# configure terminal
device(config)# vlan 70
device(config-vlan-70)# multicast6 router-port ethernet 1/1/1 to 1/1/8 ethernet 1/1/24 ethernet 1/6/24
ethernet 1/8/17
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# multicast6 static-group

Configures a static multicast listening discovery (MLD) group for a VLAN.

## Syntax

**multicast6 static-group** *ipv6-address* [ **count** *num* ] { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no multicast6 static-group** *ipv6-address* [ **count** *num* ] { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Command Default

The VLAN cannot forward multicast traffic to ports that do not receive MLD membership reports.

## Parameters

*ipv6-address*
Specifies the IPv6 address of the multicast group.

**count** *num*
Specifies a contiguous range of groups. The default is 1.

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]
Specifies a port, set of ports, or range of ports to be included in the group.

**lag** *lag-id* [ **to** *lag-id* ]
Specifies a LAG, set of LAGs, or range of LAGs to be included in the group.

## Modes

VLAN configuration mode

## Usage Guidelines

A snooping-enabled VLAN cannot forward multicast traffic to ports that do not receive MLD membership reports. To allow clients to send reports, you can configure a static group that applies to individual ports on the VLAN. The static group forwards packets to the static group ports even if they have no client membership reports.

You cannot configure a static group that applies to an entire VLAN.

The maximum number of supported static groups in a VLAN is 512, and the maximum number of supported static groups for individual ports in a VLAN is 256.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of this command removes the static group fromr the VLAN.

# Examples

This example configures on VLAN 20 a static group containing ports 1/1/3 and 1/1/5 to 1/1/7.

```
Device(config)# vlan 20
(config-vlan-20)# multicast6 static-group ff05::100 count 2 ethernet 1/1/3 ethernet 1/1/5 to 1/1/7
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# multicast6 tracking

Enables tracking and fast leave for IPv6 multicast listening discovery Version 2 (MLDv2) on VLANs.

## Syntax

**multicast6 tracking**

**no multicast6 tracking**

## Command Default

Tracking and fast leave are disabled.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of this command restores the default, that is, tracking and fast leave are disabled.

The membership tracking and fast leave features are supported for MLDv2 only. If any port or any client is not configured for MLDv2, the multicast tracking command is ignored.

## Examples

This example enables tracking and fast leave on VLAN 20.

```
Device(config)#vlan 20
Device(config-vlan-20)#multicast6 tracking
```

# multicast6 version

Configures the multicast listening discovery (MLD) version for snooping on a VLAN.

## Syntax

**multicast6 version** { **1** | **2** }

**no multicast6 version** { **1** | **2** }

## Command Default

The globally configured MLD version is configured.

## Parameters

**1**

Configures MLD Version 1.

**2**

Configures MLD Version 2.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the globally configured MLD version.

If an MLD version is specified for individual ports, these ports use that version instead of the version specified for the VLAN.

## Examples

This example specifies MLD Version 2 on VLAN 20.

```
Device(config)# vlan 20
Device(config-vlan-20)#multicast6 version 2
```

# multipath

Changes load sharing to apply to only iBGP or eBGP paths, or to support load sharing among paths from different neighboring autonomous systems.

## Syntax

**multipath** { **ebgp** | **ibgp** | **multi-as** }

**no multipath** { **ebgp** | **ibgp** | **multi-as** }

## Parameters

**ebgp**
Enables load sharing of eBGP paths only.

**ibgp**
Enables load sharing of iBGP paths only.

**multi-as**
Enables load sharing of paths from different neighboring autonomous systems.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

By default, when BGP load sharing is enabled, both iBGP and eBGP paths are eligible for load sharing, while paths from different neighboring autonomous systems are not.

The **no** form of the command restores the default.

## Examples

The following example changes load sharing to apply to iBGP paths in the IPv4 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# multipath ibgp
```

The following example enables load sharing of paths from different neighboring autonomous systems in the IPv6 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# multipath multi-as
```

The following example changes load sharing to apply to eBGP paths in a nondefault VRF instance in the IPv4 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# multipath ebgp
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# multi-spx-lag

Changes both ends of live SPX ports to form an SPX LAG.

## Syntax

**multi-spx-lag** *port-list1* **and** *port-list2*

**no multi-spx-lag** *port-list1* **and** *port-list2*

## Command Default

Individual SPX links, rather than an SPX LAG, exist before the command is executed.

## Parameters

*port-list1*
Designates the ports that form one end of the SPX LAG.

*port-list2*
Designates the ports that form the other end of the SPX LAG.

## Modes

CB configuration mode

SPX unit configuration mode

## Usage Guidelines

The **no** form of the command removes SPX LAGs in a live link.

The **no multi-spx-port** and **no multi-spx-lag** commands provide the only way to change a PE ring to two chains or one chain without physically removing the cable. The removed spx-ports and spx-lags are brought down and disabled in configuration.

The system blocks the **multi-spx-lag** command if executing the command would make any PE unreachable.

This command changes both ends of a live CB-to-PE or PE-to-PE link at the same time to form an SPX LAG. Using the command avoids generating transit port-to-LAG connections and breaking internal communication.

Both **multi-spx-port** and **multi-spx-lag** are available in CB configuration mode and SPX unit configuration modes.

## Examples

The following example creates a live LAG in CB configuration mode between the designated ports on CB unit 3 and PE unit 24.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# multi-spx-lag 3/1/6 3/1/8 and 24/3/1 24/4/1
spx-port 24/4/1 is replaced by spx-lag 24/3/1 24/4/1.
spx-port 3/1/8  is replaced by spx-lag 3/1/6 3/1/8.
```

The following example creates a live LAG between PE unit 17 and PE unit 24 as part of configuring PE unit 24 from the CB in SPX unit configuration mode. In this case, the command could be configured either under spx unit 17 or spx unit 24.

```
device# configure terminal
device(config)# spx unit 24
device(config-spx-unit-24)# multi-spx-lag 24/2/1 to 24/2/2 and 17/2/1 to 17/2/2
spx-port 17/2/1 is replaced by spx-lag 17/2/1 to 17/2/2.
spx-port 24/2/1 is replaced by spx-lag 24/2/1 to 24/2/2.
```

The following example uses the **no multi-spx-lag** command to remove an SPX LAG between PE units 26 and 27 in a live system and disables related ports on both units.

```
ICX7750-20Q Router# show spx

… snipped


    active       standby
     +---+        +---+        +---+
 =2/1| 2 |2/4--2/4| 3 |2/1==2/1| 1 |2/4=
  |  +---+        +---+        +---+   |
  |                                     |
  |-------------------------------------|
            +-----+        +-----+        +-----+        +-----+        +-----+
  1/1/5==3/1| 17 |2/2--2/2| 27 |2/3==2/1| 26 |2/5==2/5| 25 |2/1==2/5| 24 |2/1=
            +-----+        +-----+        +-----+        +-----+        +-----+   |
                                                                                   |
                                                                          +-----+  |
                                                                2/1/9--4/1| 23 |2/1=
                                                                          +-----+



ICX7750-20Q Router# config terminal
ICX7750-20Q Router(config)# spx unit 27
ICX7750-20Q Router(config-spx-unit-27)# no multi-spx-lag 27/2/3 to 27/2/4 and 26/2/1 to 26/2/2
Bring down ports: 26/2/1 to 26/2/2 27/2/3 to 27/2/4
Wait for 20 sec before removing spx-lags. These ports will be disabled in configuration.
spx-lag 26/2/1 to 26/2/2 is removed
spx-lag 27/2/3 to 27/2/4 is removed

ICX7750-20Q Router(config-spx-unit-27)#
ICX7750-20Q Router(config-spx-unit-27)# show spx

… snipped


    active       standby
     +---+        +---+        +---+
 =2/1| 2 |2/4--2/4| 3 |2/1==2/1| 1 |2/4=
  |  +---+        +---+        +---+   |
  |                                     |
  |-------------------------------------|
            +-----+        +-----+
  1/1/5==3/1| 17 |2/2--2/2| 27 |
            +-----+        +-----+

            +-----+        +-----+        +-----+        +-----+
  2/1/9--4/1| 23 |2/1==2/1| 24 |2/5==2/1| 25 |2/5==2/5| 26 |
            +-----+        +-----+        +-----+        +-----+
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |

| Release version | Command history |
|---|---|
| 08.0.80 | The no form of the command became available. |

# multi-spx-port

Changes both ends of a live SPX LAG into SPX ports.

## Syntax

**multi-spx-port** *port1* **and** *port2*

**no multi-spx-port** *port1* **and** *port2*

## Command Default

By default, no SPX LAG exists. The command is used to transform an SPX LAG that has been previously configured.

## Parameters

*port1*

Designates the interface on one side of the live SPX LAG.

*port2*

Designates the interface on the other side of the live SPX LAG.

## Modes

CB configuration mode

SPX unit configuration mode

## Usage Guidelines

The **no** form of the command removes SPX ports from a live link.

The **no multi-spx-port** and **no multi-spx-lag** commands provide the only way to change a PE ring to two chains or one chain without physically removing the cable. The removed spx-ports and spx-lags are brought down, changed to data ports, and disabled in configuration.

The system blocks the **multi-spx-port** command if the command would make any PE unreachable.

The command can be used to change an SPX LAG to an SPX port when the SPX LAG is active. The command can be applied to a link between a CB and a PE unit or between two PE units.

# Examples

The following example show a live SPX lag being created and then converted into a live SPX port between CB unit 3 and PE unit 24.

```
device# configure terminal
device(config)# spx cb-enable
System is now in 802.1br control bridge (CB) mode.
device(config)# spx cb-config
device(config-spx-cb)# multi-spx-lag 3/1/6 3/1/8
device(config-spx-cb)# multi-spx-lag 24/3/1 24/4/1
.
.
.
device(config-spx-cb)# end
device#
```

Once the SPX LAG created in the previous code block is live, it can be modified. Here, it is replaced on a live system by two SPX ports.

```
device# configure terminal
device(config)# spx cb-config
device(config-spx-cb)# multi-spx-port 3/1/8 and 24/4/1
spx-lag 3/1/6 3/1/8  is replaced by spx-port 3/1/8.
spx-lag 24/3/1 24/4/1 is replaced by spx-port 24/4/1.
```

The following example creates a live link between PE unit 17 and PE unit 24.

```
device# configure terminal
device(config)# spx cb-config
device(config-spx-cb)# multi-spx-port 24/2/1 and 17/2/1
spx-lag 17/2/1 to 17/2/2 is replaced by spx-port 17/2/1.
spx-lag 24/2/1 to 24/2/2 is replaced by spx-port 24/2/1.
```

The following example removes a live link using the **no multi-spx-port** command. The example removes the SPX ports between PE units 24 and 25.

```
ICX7750-20Q Router(config)# spx cb-configure
ICX7750-20Q Router(config-spx-cb)# no multi-spx-port 24/2/1 and 25/2/3
Bring down 24/2/1 and 25/2/3
Wait for 20 sec before removing spx-ports. These ports will be disabled in configuration.
ICX7750-20Q Router(config-spx-cb)# end
ICX7750-20Q Router# show spx
T=22h55m20.8: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7750-20QXG active  cc4e.2439.2a80 128 local   Ready
2  S ICX7750-20QXG standby cc4e.2439.3700   0 remote  Ready
3  S ICX7750-48XGC member  cc4e.2439.1680   0 remote  Ready
17 S ICX7150-C12P  spx-pe  609c.9fbc.bf26 N/A remote  Ready
18 S ICX7150-48ZP  spx-pe  609c.9fee.4320 N/A remote  Ready
23 S ICX7450-48GF  spx-pe  cc4e.246c.f850 N/A remote  Ready
24 S ICX7450-48P   spx-pe  609c.9fc2.3090 N/A remote  Ready
25 S ICX7250-24    spx-pe  cc4e.24df.0ade N/A remote  Ready

    standby      active
     +---+        +---+        +---+
  3/1| 2 |2/1==2/1| 1 |3/1==3/1| 3 |2/1
     +---+        +---+        +---+
            +----+
 3/1/41==1/7| 17 |
            +----+

            +----+
 3/1/17==1/1| 18 |
            +----+

            +----+        +----+
  2/3/6--4/1| 23 |3/1--4/1| 24 |2/1-
            +----+        +----+

            +----+
 3/1/15--1/1| 25 |2/3-
            +----+

Standby u2 - protocols ready, can failover
ICX7750-20Q Router#   spx-port 24/2/1 is removed
spx-port 25/2/3 is removed
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.80 | The no form of the command became available. |

# multi-stack-port

Converts both ends of a trunked link between stacking trunks in a traditional stack to links between untrunked ports.

## Syntax

**multi-stack-port** *stack-unit/slot/port* **and** *stack-unit/slot/port*

**no multi-stack-port** *stack-unit/slot/port* **and** *stack-unit/slot/port*

## Parameters

*stack-unit*
: Specifies the stack unit ID.

*slot*
: Specifies the slot number.

*port*
: Specifies the port number in the slot.

## Modes

Stack unit configuration mode

## Usage Guidelines

The **no** form of the command removes the stack-ports.

Use this command only when the trunk is live.

Only primary stacking ports can be designated in the command.

## Examples

The following example converts the stacking trunk between stack unit 3 and stack unit 4 to a link between untrunked ports.

```
device# configure terminal
device(config)# stack unit 1
device(config-unit-3)# multi-stack-port 3/2/1 and 4/2/1
```

## History

| Release | Command History |
|---|---|
| 08.0.00a | This command was introduced. It replaces the **multi-port** command. |

# multi-stack-trunk

Creates both ends of a multi-port stacking trunk on a live stack.

## Syntax

**multi-stack-trunk** *stack-unit/slot/port* **to** *stack-unit/slot/port* **and** *stack-unit/slot/port* **to** *stack-unit/slot/port*

**no multi-stack-trunk** *stack-unit/slot/port* **to** *stack-unit/slot/port* **and** *stack-unit/slot/port* **to** *stack-unit/slot/port*

## Parameters

*stack-unit*
: Specifies the stack unit ID.

*slot*
: Specifies the slot number.

*port*
: Specifies the port number in the slot.

## Modes

Stack unit configuration mode

## Usage Guidelines

Use the command only when the trunk is live.

The first port in a stack trunk must be an odd-numbered primary port, for example, 3/2/1.

The **no** form of this command removes the trunk configuration.

## Examples

The following example converts two non-trunked links between stack unit 3 and stack unit 4 into a stacking trunk. The stacking trunk connects ports 3/2/1 and 3/2/2 on stack unit 3 and ports 4/2/1 and 4/2/2 on stack unit 4.

```
device# configure terminal
device(config)# stack unit 1
device(config-unit-1)# multi-stack-trunk 3/2/1 to 3/2/2 and 4/2/1 to 4/2/2
```

## History

| Release | Command History |
|---|---|
| FastIron release 08.0.00a | This command was introduced. This command replaces the **multi-trunk** command. |

# Commands N

## name (MRP)

Configures the name for the metro ring.

### Syntax

**name** *string*

**no name** *string*

### Command Default

Metro ring names are not configured.

### Parameters

*string*

Specifies the name for the metro ring. The name is an ASCII string and can be up to 64 characters in length and include blank spaces.

### Modes

MRP configuration mode

### Usage Guidelines

The name is optional for a metro ring. If you use a name that has blank spaces, enclose the name in double quotation marks, for example, "Customer A".

The **no** form of the command removes the name for the metro ring.

### Examples

The following example configures the name for a metro ring.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-1)# name CustomerA
```

# nbr-timeout

Configures the interval after which a PIM device considers a neighbor to be absent.

## Syntax

**nbr-timeout** *seconds*

**no nbr-timeout** *seconds*

## Command Default

The timeout interval is 105 seconds.

## Parameters

*seconds*

Specifies the interval, in seconds. The range is 35 through 65535 seconds. The default is 105 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the default timeout interval, 105 seconds.

You should set the interval to be not less than 3.5 times the hello timer value.

## Examples

This example configures a PIM neighbor timeout value of 360 seconds on all ports on a device operating with PIM.

```
Device(config)# router pim
Device(config-pim-router)# nbr-timeout 360
```

# neighbor (RIP)

Configures RIP neighbor filter to specify RIP routes to be learned and advertised.

## Syntax

**neighbor** *filter-num* { **permit** | **deny** } { **any** | *ip-address* }

**no neighbor** *filter-num* { **permit** | **deny** } { **any** | *ip-address* }

## Command Default

Initially, by default, the device learns all RIP routes from all neighbors and advertises all routes to all neighbors. Once you have defined a filter that permits learning from a RIP neighbor, the default changes so that the device denies all other RIP neighbors except those specified.

## Parameters

*filter-num*
: Filter index number, a decimal value from 1 through 64.

**permit**
: Allows routes to be learned and advertised for designated IP address or for any IP address, depending on configuration.

**deny**
: Prevents routes from being learned or advertised for designated IP address or for any IP address, depending on configuration.

**any**
: Indicates configured action is to be applied to all IP addresses.

*ip-address*
: Specifies an IP address to which the filter applies.

## Modes

RIP router configuration mode.

## Usage Guidelines

The **no** form of the command deactivates the filter.

You may require more than one filter to obtain the results you want. For example, if you create a filter to allow or deny a specific IP address, you must create additional filters to allow route learning and advertisement for any other IP addresses.

To avoid conflicting actions, give the filter with the highest priority the highest filter number. Typically, you would add the priority filter last. For example, If you want to deny only one IP address, you must create a second filter with a higher number (priority) to allow any others.

# Examples

The following example configures the RIP router so that no RIP routes are learned or advertised for any neighbor.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# neighbor 1 deny any
```

The following example configures the RIP router to learn and advertise routes for all neighbors except neighboring IP address 10.70.12.104. Note the second filter is required and must have a higher filter number.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# neighbor 2 deny 10.70.12.104
device(config-rip-router)# neighbor 64 permit any
```

# neighbor activate

Enables the exchange of information with BGP neighbors and peer groups.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **activate**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **activate**

## Command Default

Enabling address exchange for the IPv6 address family is disabled.

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies a peer group.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

The **no** form of the command disables the exchange of an address with a BGP neighbor or peer group.

## Examples

The following example establishes a BGP session with a neighbor with the IPv6 address 2001:2018:8192::125.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 activate
```

The following example establishes a BGP session with a neighbor with the IPv6 address 2001:2018:8192::125 for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 activate
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor advertisement-interval

Enables changes to the interval over which a specified neighbor or peer group holds route updates before forwarding them.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **advertisement-interval** *seconds*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **advertisement-interval**

## Parameters

*ip-address*
IPv4 address of the neighbor.

*ipv6-address*
IPv6 address of the neighbor.

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

*seconds*
Range is from 0 through 3600. The default is 0.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the default interval.

## Examples

The following example changes the BGP4 advertisement interval from the default to 60 seconds.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 advertisement-interval 60
```

The following example changes the BGP4+ advertisement interval from the default for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 advertisement-interval 60
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor allowas-in

Disables the AS_PATH check function for routes learned from a specified location so that BGP does not reject routes that contain the recipient BGP speaker's AS number.

## Syntax

**neighbor** {*ip-address* | *ipv6-address* | *peer-group-name* } **neighbor allowas-in** *number*

**no neighbor allowas-in** {*ip-address* | *ipv6-address* | *peer-group-name* } **neighbor allowas-in** *number*

## Command Default

The AS_PATH check function is enabled and any route whose path contains the speaker's AS number is rejected as a loop.

## Parameters

*ip-address*
: Specifies the IP address of the neighbor.

*ipv6-address*
: Specifies the IPv6 address of the neighbor.

*peer-group-name*
: Specifies a peer group.

*number*
: Specifies the number of times that the AS path of a received route may contain the recipient BGP speaker's AS number and still be accepted. Valid values range from 1 through 10.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

The **no** form of the command re-enables the AS_PATH check function.

# Examples

The following example specifies that the AS path of a received route may contain the recipient BGP speaker's AS number three times and still be accepted.

```
device#configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 allowas-in 3
```

# neighbor as-override

Replaces the autonomous system number (ASN) of the originating device with the ASN of the sending BGP device.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **as-override**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **as-override**

## Parameters

*ip-address*
IPv4 address of the neighbor.

*ipv6-address*
IPv6 address of the neighbor.

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

## Modes

BGP configuration mode

## Usage Guidelines

BGP loop prevention verifies the ASN in the AS path. If the receiving router sees its own ASN in the AS path of the received BGP packet, the packet is dropped. The receiving router assumes that the packet originated from its own AS and has reached the place of origination. This can be a significant problem if the same ASN is used among various sites, preventing sites with identical ASNs from being linked by another ASN. In this case, routing updates are dropped when another site receives them.

## Examples

The following example replaces the ASN globally.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 as-override
```

# neighbor capability as4

Enables or disables support for 4-byte autonomous system numbers (ASNs) at the neighbor or peer-group level.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **capability as4** [ **disable** | **enable** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **capability as4** [ **disable** | **enable** ]

## Command Default

4-byte ASNs are disabled by default.

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

**disable**
Disables 4-byte numbering.

**enable**
Enables 4-byte numbering.

## Modes

BGP configuration mode

## Usage Guidelines

4-byte ASNs are first considered at the neighbor, then at the peer group, and finally at the global level.

The **disable** keyword or the **no** form of the command removes all neighbor capability for 4-byte ASNs.

## Examples

The following example enables 4-byte ASNs globally.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 capability as4 enable
```

# neighbor capability orf prefixlist

Advertises outbound route filter (ORF) capabilities to peer routers.

## Syntax

**neighbor** { *ip_address* | *ipv6_address* | *peer-group-name* } **capability orf prefixlist** [ **receive** | **send** ]

**no neighbor** { *ip_address* | *ipv6_address* | *peer-group-name* } **capability orf prefixlist** [ **receive** | **send** ]

## Command Default

ORF capabilities are not advertised to a peer device.

## Parameters

*ip_address*
: Specifies the IPv4 address of the neighbor.

*ipv6_address*
: Specifies the IPv6 address of the neighbor.

*peer-group-name*
: Specifies a peer group.

**receive**
: Enables the ORF prefix list capability in receive mode.

**send**
: Enables the ORF prefix list capability in send mode.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

The **no** form of the command disables ORF capabilities.

# Examples

The following example advertises the ORF send capability to a neighbor with the IP address 10.11.12.13.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 capability orf prefixlist send
```

The following example advertises the ORF receive capability to a neighbor with the IPv6 address 2001:2018:8192::125.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 capability orf prefixlist receive
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor default-originate

Configures the device to send the default route 0.0.0.0 to a neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **default-originate** [ **route-map** *map-name* ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **default-originate** [ **route-map** *map-name* ]

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name.

**route-map**
Optionally injects the default route conditionally, depending on the match conditions in the route map.

*map-name*
Specifies a route map.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example sends the default route to a BGP4 neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 default-originate route-map myroutemap
```

The following example sends the default route for a BGP4+ neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 default-originate route-map myroutemap22
```

# neighbor description

Specifies a name for a neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **description** *string*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **description**

## Parameters

*ip-address*
IPv4 address of the neighbor.

*ipv6-address*
IPv6 address of the neighbor.

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

**description** *string*
Specifies the name of the neighbor, an alphanumeric string up to 220 characters long.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes the name.

## Examples

The following example specifies a BGP4 neighbor name.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 description mygoodneighbor
```

The following example specifies a BGP4+ neighbor name for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 default-originate route-map myroutemap
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor ebgp-btsh

Enables BGP time to live (TTL) security hack protection (BTSH) for eBGP.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **ebgp-btsh**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **ebgp-btsh**

## Command Default

Disabled.

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies a peer group.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations. To maximize the effectiveness of this feature, the **neighbor ebgp-btsh** command should be executed on each participating device.

The **neighbor ebgp-btsh** command is supported for both directly connected peering sessions and multihop eBGP peering sessions. When the **neighbor ebgp-btsh** command is used, BGP control packets sent by the device to a neighbor have a TTL value of 255. In addition, the device expects the BGP control packets received from the neighbor to have a TTL value of either 254 or 255. For multihop peers, the device expects the TTL for BGP control packets received from the neighbor to be greater than or equal to 255, minus the configured number of hops to the neighbor. If the BGP control packets received from the neighbor do not have the anticipated value, the device drops them.

The **no** form of the command disables BTSH for eBGP.

# Examples

The following example enables GTSM between a device and a neighbor with the IP address 10.10.10.1.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.1.1.1 ebgp-btsh
```

The following example enables GTSM between a device and a neighbor with the IPv6 address 2001:2018:8192::125.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 ebgp-btsh
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor ebgp-multihop

Allows eBGP neighbors that are not on directly connected networks and sets an optional maximum hop count.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **ebgp-multihop** [ *max-hop-count* ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **ebgp-multihop**

## Parameters

*ip-address*
IPv4 address of the neighbor

*ipv6-address*
IPv6 address of the neighbor

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

*max-hop-count*
Maximum hop count. Range is from 1 through 255.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Examples

The following example enables eBGP multihop and sets the maximum hop count to 20.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 ebgp-multihop 20
```

The following example enables BGP4+ eBGP multihop for VRF instance "red" and sets the maximum hop count to 40.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 ebgp-multihop 40
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor enforce-first-as

Ensures that a device requires the first ASN listed in the AS_SEQUENCE field of an AS path-update message from eBGP neighbors to be the ASN of the neighbor that sent the update.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **enforce-first-as** [ **disable** | **enable** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **enforce-first-as** [ **disable** | **enable** ]

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

**disable**
Disables this feature.

**enable**
Enables this feature.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command disables this requirement globally for the device.

## Examples

The following example enables the enforce-first-as feature for a specified neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 enforce-first-as enable
```

# neighbor filter-list

Specifies a filter list to be applied to updates from or to the specified neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **filter-list** *ip-prefix-list-name* { **in** | **out** }

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **filter-list** *ip-prefix-list-name* { **in** | **out** }

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies theIPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*ip-prefix-list-name*
Specifies the name of the filter list.

**in**
Specifies that the list is applied on updates received from the neighbor.

**out**
Specifies that the list is applied on updates sent to the neighbor.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example specifies that filter list “myfilterlist” be applied to updates to a neighbor with the IP address 10.11.12.13.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 filter-list myfilterlist out
```

The following example specifies that filter list “2” be applied to updates from a neighbor with the IPv6 address 2001:2018:8192::125.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 filter-list 2 in
```

# neighbor local-as

Causes the device to prepend the local autonomous system number (ASN) automatically to routes received from an eBGP peer.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **local-as** *num* [ **no-prepend** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **local-as** *num* [ **no-prepend** ]

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*num*
Specifies the local ASN. Range is from 1 through 4294967295.

**no-prepend**
Causes the device to stop prepending the selected ASN.

## Modes

BGP configuration mode

## Usage Guidelines

The **no** form of the command removes the local ASN.

## Examples

The following example ensures that a device prepends the local ASN.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 local-as 100
```

The following example stops the device from prepending the selected ASN.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 local-as 100 no-prepend
```

# neighbor maxas-limit in

Causes the device to discard routes received in UPDATE messages if those routes exceed a maximum AS path length.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **maxas-limit in** { *num* | **disable** }

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **maxas-limit in**

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name.

*num*
Specifies the maximum length of the AS path. Valid values range from 0 through 300. The default is 300.

**disable**
Prevents a neighbor from inheriting the configuration from the peer group or global configuration and instead uses the default system value.

## Modes

BGP configuration mode

## Examples

The following example changes the length of the maximum allowed AS path length from the default.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 maxas-limit in 200
```

The following example prevents a neighbor from inheriting the configuration from the peer group or global configuration and instead use the default system value.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 2001:2018:8192::125 maxas-limit in disable
```

# neighbor maximum-prefix

Specifies the maximum number of IP network prefixes (routes) that can be learned from a specified neighbor or peer group.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **maximum-prefix** *num* [ *threshold* ] [ **teardown** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **maximum-prefix** *num* [ *threshold* ] [ **teardown** ]

## Parameters

*ip-address*
: Specifies the IPv4 address of the neighbor.

*ipv6-address*
: Specifies the IPv6 address of the neighbor.

*peer-group-name*
: Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*num*
: Specifies the maximum number of IP prefixes that can be learned. Range is from 0 through 4294967295. Default is 0 (unlimited).

*threshold*
: Specifies the percentage of the value specified by *num* that causes a syslog message to be generated. Range is from 1 through 100. Default is 100.

**teardown**
: Tears down the neighbor session if the maximum number of IP prefixes is exceeded.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example sets the maximum number of prefixes that will be accepted from the neighbor with the IP address 10.11.12.13 to 100000, and sets the threshold value to 80%.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)#  neighbor 10.11.12.13 maximum-prefix 100000 threshold 80
```

# neighbor next-hop-self

Causes the device to list itself as the next hop in updates that are sent to the specified neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **next-hop-self** [ **always** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **next-hop-self** [ **always** ]

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

**always**
Enables this feature for route reflector (RR) routes.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command disables this feature.

## Examples

The following example causes all updates destined for the neighbor with the IP address 10.11.12.13 to advertise this device as the next hop.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 next-hop-self
```

The following example, for the VRF instance "red," causes all updates destined for the neighbor with the IPv6 address 2001:2018:8192::125 to advertise this device as the next hop.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 next-hop-self
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor password

Specifies an MD5 password for securing sessions between the device and a neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **password** *string*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **password**

## Command Default

No password is set.

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*string*
Password of up to 63 characters in length that can contain any alphanumeric character.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes a configured MD5 password.

## Examples

The following example specifies a password for securing sessions with a specified neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 password s0M3P@55W0Rd
```

The following BGP4+ example, for VRF instance "red," specifies a password for securing sessions with a specified neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv4u-vrf)# neighbor 2001:2018:8192::125 password s0M3P@55W0Rd
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor peer-group

Configures a BGP neighbor to be a member of a peer group.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* } **peer-group** *string*

**no neighbor** { *ip-address* | *ipv6-address* } **peer-group** *string*

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor.

**peer-group** *string*
Specifies the name of a BGP peer group. The name can be up to 63 characters in length and can be composed of any alphanumeric character.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes a neighbor from the peer group.

## Examples

The following example assigns a specified neighbor to a peer group called “mypeergroup1”.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 peer-group mypeergroup1
```

The following BGP4+ example, for VRF instance "red," assigns a specified neighbor to a peer group called "mypeergroup1".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv4u)# neighbor 2001:2018:8192::125 peer-group mypeergroup1
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor prefix-list

Filters the outgoing and incoming route updates to or from a particular BGP neighbor according to IP address and mask length.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **prefix-list** *string* { **in** | **out** }

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **prefix-list** *string* { **in** | **out** }

## Parameters

*ip-address*
: Specifies the IPv4 address of the neighbor

*ipv6-address*
: Specifies the IPv6 address of the neighbor

*peer-group-name*
: Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

*string*
: Specifies the name of the prefix list.

**in**
: Applies the filter in incoming routes.

**out**
: Applies the filter in outgoing routes.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example applies the prefix list "myprefixlist" to incoming advertisements to neighbor 10.11.12.13.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 prefix-list myprefixlist in
```

The following example applies the prefix list "myprefixlist" to outgoing advertisements to neighbor 2001:2018:8192::125.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# neighbor 2001:2018:8192::125 prefix-list myprefixlist out
```

# Related Commands

prefix-listprefix-listprefix-list

# neighbor remote-as

Specifies the autonomous system (AS) in which a remote neighbor resides.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **remote-as** *num*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **remote-as**

## Command Default

No AS is specified.

## Parameters

*ip-address*
IPv4 address of the neighbor

*ipv6-address*
IPv6 address of the neighbor

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

*num*
Remote AS number (ASN). Range is from 1 through 4294967295.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command removes the neighbor from the AS.

## Examples

The following example specifies AS 100 for a neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 remote-as 100
```

The following BGP4+ example, for VRF instance "red," specifies AS 100 for a neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 remote-as 100
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor remove-private-as

Configures a device to remove private autonomous system numbers (ASNs) from UPDATE messages that the device sends to a neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **remove-private-as**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **remove-private-as**

## Parameters

*ip-address*
IPv4 address of the neighbor

*ipv6-address*
IPv6 address of the neighbor

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The device will remove ASNs 64512 through 65535 (the well-known BGP4 private ASNs) from the AS-path attribute in UPDATE messages that the device sends to a neighbor.

The **no** form of the command restores the default so that private ASNs are not removed from UPDATE messages sent to a neighbor by a device.

## Examples

The following example removes private ASNs globally.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 remove-private-as
```

The following example removes private ASNs for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 remove-private-as
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor route-map

Filters the outgoing and incoming route updates to or from a particular BGP neighbor according to a set of attributes defined in a route map.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **route-map** { **in** *string* | **out** *string* }

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **route-map** { **in** *string* | **out** *string* }

## Parameters

*ip-address*
: Specifies the IPv4 address of the neighbor

*ipv6-address*
: Specifies the IPv6 address of the neighbor

*peer-group-name*
: Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

**in**
: Applies the filter on incoming routes.

*string*
: Name of the route map.

**out**
: Applies the filter on outgoing routes.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example applies a route map named "myroutemap" to an outgoing route from 10.11.12.13.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 route-map myroutemap out
```

# neighbor route-reflector-client

Configures a neighbor to be a route-reflector client.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **route-reflector-client**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **route-reflector-client**

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor

*ipv6-address*
Specifies the IPv6 address of the neighbor

*peer-group-name*
Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

Use this command on a host device to configure a neighbor to be a route-reflector client. Once configured, the host device from which the configuration is made acts as a route-reflector server.

## Examples

The following example configures a neighbor to be a route-reflector client.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 route-reflector-client
```

# neighbor send-community

Enables sending the community attribute in updates to the specified BGP neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **send-community** [ **both** | **extended** | **standard** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **send-community** [ **both** | **extended** | **standard** ]

## Command Default

The device does not send community attributes.

## Parameters

*ip-address*
: Specifies the IPv4 address of the neighbor

*ipv6-address*
: Specifies the IPv6 address of the neighbor

*peer-group-name*
: Specifies the peer group name configured by the **neighbor** *peer-group-name* command.

**both**
: Sends both standard and extended attributes.

**extended**
: Sends extended attributes.

**standard**
: Sends standard attributes.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example sends standard community attributes to a neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 send-community standard
```

# neighbor shutdown

Causes a device to shut down the session administratively with its BGP neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **shutdown** [ **generate-rib-out** ]

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **shutdown** [ **generate-rib-out** ]

## Parameters

*ip-address*
IPv4 address of the neighbor

*ipv6-address*
IPv6 address of the neighbor

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

**generate-rib-out**
When a peer is put into the shutdown state, Routing Information Base (RIB) outbound routes are not produced for that peer. Use this option to produce those routes.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Shutting down a session lets you configure the neighbor and save the configuration without the need to establish a session with that neighbor.

## Examples

The following example causes a device to shut down the session administratively with its neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 shutdown
```

The following example causes a device to shut down the session administratively with its neighbor and generate RIB outbound routes for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 shutdown generate-rib-out
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor soft-reconfiguration inbound

Stores all the route updates received from a BGP neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **soft-reconfiguration inbound**

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **soft-reconfiguration inbound**

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor

*ipv6-address*
Specifies the IPv6 address of the neighbor

*peer-group-name*
Specifies the peer group name.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Soft reconfiguration stores all the route updates received from a neighbor. If you request a soft reset of inbound routes, the software compares the policies against the stored route updates, instead of requesting the neighbor's BGP4 or BGP4+ route table or resetting the session with the neighbor.

## Examples

The following example globally stores route updates from a BGP4 neighbor.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 soft-configuration inbound
```

The following example stores route updates from a BGP4+ neighbor for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 2001:2018:8192::125 soft-configuration inbound
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor timers

Specifies how frequently a device sends KEEPALIVE messages to its BGP neighbors, as well as how long the device waits for KEEPALIVE or UPDATE messages before concluding that a neighbor is dead.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **timers keep-alive** *keepalive_interval* **hold-time** *holdtime_interval*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **timers keep-alive** *keepalive_interval* **hold-time** *holdtime_interval*

## Parameters

*ip-address*
IPv4 address of the neighbor

*ipv6-address*
IPv6 address of the neighbor

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

**keep-alive** *keepalive_interval*
Frequency (in seconds) with which a device sends keepalive messages to a peer. Range is from 0 through 65535 seconds. The default is 60.

**hold-time** *holdtime_interval*
Interval in seconds that a device waits to receive a keepalive message from a peer before declaring that peer dead. Range is from 0 through 65535 seconds. The default is 180.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

The **no** form of the command restores the defaults.

## Examples

The following example sets the keepalive timer for a device to 120 seconds and the hold-timer to 360 seconds.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 timers keep-alive 120 hold-time 360
```

The following example sets the keepalive timer to 120 seconds and the hold-timer to 360 seconds for VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# neighbor 10.11.12.13 timers keep-alive 120 hold-time 360
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# neighbor update-source

Configures the device to communicate with a neighbor through a specified interface.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **update-source** { *ip-address* | *ipv6-address* | **ethernet** *unit/slot/port* | **lag** *lag-id* | **loopback** *num* | **ve** *vlan_id* }

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **update-source** { *ip-address* | *ipv6-address* | **ethernet** *unit/slot/port* | **lag** *lag-id* | **loopback** *num* | **ve** *vlan_id* }

## Parameters

*ip-address*
IPv4 address of the neighbor

*ipv6-address*
IPv6 address of the neighbor

*peer-group-name*
Peer group name configured by the **neighbor** *peer-group-name* command.

*ip-address*
IP address of the update source.

*ipv6-address*
IPv6 address of the update source.

**ethernet** *unit/slot/port*
Specifies the physical interface.

**lag** *lag-id*
Specifies a LAG virtual interface.

**loopback** *num*
Specifies a loopback interface.

**ve** *vlan_id*
Specifies a virtual Ethernet VLAN interface.

## Modes

BGP configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults.

# Examples

This example configures the device globally to communicate with a neighbor through the specified IPv4 address and port.

```
device#configure terminal
device#(config)# router bgp
device(config-bgp)# neighbor 10.11.12.13 update-source ethernet 5/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |
| 08.0.61 | The command was modified to include lag-id options. |

# neighbor weight

Specifies a weight that the device will add to routes that are received from the specified BGP neighbor.

## Syntax

**neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **weight** *num*

**no neighbor** { *ip-address* | *ipv6-address* | *peer-group-name* } **weight** *num*

## Parameters

*ip-address*
Specifies the IPv4 address of the neighbor.

*ipv6-address*
Specifies the IPv6 address of the neighbor

*peer-group-name*
Specifies the name of the peer group.

*num*
Specifies a value. Valid values range from 1 through 65535. The default is 0.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

BGP prefers larger weights over smaller weights.

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example changes the weight from the default.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# neighbor 10.11.12.13 weight 100
```

# netbios-name-server

Specifies the IP address of a NetBIOS WINS server or servers available to Microsoft DHCP clients.

## Syntax

**netbios-name-server** *address* [*address2,address3* ]

## Parameters

*address*

Specifies the IP address of the NetBIOS WINS server.

## Modes

DHCP server pool configuration mode.

## Examples

The following example specifies the IP address of a NetBIOS WINS server.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# netbios-name-server 192.168.1.55
```

# netbios-proto

Configures the NetBIOS protocol-based VLAN and enters NetBIOS protocol VLAN configuration mode.

## Syntax

**netbios-proto** [ **name** *string* ]

**no netbios-proto** [ **name** *string* ]

## Command Default

An NetBIOS protocol-based VLAN is not configured.

## Parameters

**name** *string*

Specifies the name of the NetBIOS protocol configuration. The name can be up to 32 characters in length.

## Modes

VLAN configuration mode

IP protocol VLAN configuration mode

IPX protocol VLAN configuration mode

IPv6 protocol VLAN configuration mode

DECnet protocol VLAN configuration mode

AppleTalk protocol VLAN configuration mode

Other protocol VLAN configuration mode

## Usage Guidelines

The **no** form of the command disables the NetBIOS protocol-based VLANs.

## Examples

The following example shows how to configure the NetBIOS protocol-based VLAN.

```
device(config)# ipx-proto name Brown
device(config-vlan-ipx-proto)# netbios-proto name proto1
device(config-vlan-netbios-proto)# no dynamic
```

# network

Configures the device to advertise a network.

## Syntax

**network** *network/mask* [ **backdoor** | **route-map** *map-name* | **weight** *num* ]

**no network** *network/mask* [ **backdoor** | **route-map** *map-name* | **weight** *num* ]

## Command Default

No network is advertised.

## Parameters

*network/mask*
Network and mask in CIDR notation.

**backdoor**
Changes administrative distance of the route to this network from the EBGP administrative distance (the default is 20) to the local BGP4 weight (the default is 200), tagging the route as a backdoor route.

**route-map** *map-name*
Specifies a route map with which to set or change BGP4 attributes for the network to be advertised.

**weight***num*
Specifies a weight to be added to routes to this network. Range is 0 through 65535. The default is 0.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example imports the IPv4 network 10.11.12.12/30 into the route map "myroutemap".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# network 10.11.12.13/30 route-map myroutemap
```

The following example imports the IPv6 prefix 2001:db8::/32 into the BGP4+ database and sets a weight of 300.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# network 2001:db8::/32 weight 300
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# network (dhcp)

Configures the subnet network and mask of the DHCP address pool.

## Syntax

**network** *subnet/mask*

## Parameters

*subnet/mask*

Specifies the subnet network and mask of the address pool.

## Modes

DHCP server pool configuration mode

## Examples

The following command specifies the subnet network and mask of the DHCP address pool.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# network 10.2.3.44/24
```

# next-bootstrap-server

Specifies the IP address of the next server the client should use for bootup.

## Syntax

**next-bootstrap-server** *ip-address*

## Parameters

*ip-address*

Specifies the IP address of the next bootstrap server.

## Modes

DHCP server pool configuration mode.

## Examples

The following example specifies the next bootstrap server.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# next-bootstrap-server 10.2.5.44
```

# next-hop-enable-default

Configures the device to use the default route as the next hop.

## Syntax

**next-hop-enable-default**

**no next-hop-enable-default**

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

## Examples

The following example configures the device to use the default route as the next hop for the IPv4 unicast address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# next-hop-enable-default
```

The following example configures the device to use the default route as the next hop for the IPv6 unicast address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# next-hop-enable-default
```

# next-hop-recursion

Enables BGP recursive next-hop lookups.

## Syntax

**next-hop-recursion**

**no next-hop-recursion**

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

If the BGP next hop is not the immediate next hop, a recursive route lookup in the IP routing information base (RIB) is needed. With recursion, a second routing lookup is required to resolve the exit path for destination traffic. Use this command to enable recursive next-hop lookups.

## Examples

The following example enables recursive next-hop lookups for BGP4 for the IPv4 unicast address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# next-hop-recursion
```

The following example enables recursive next-hop lookups for the IPv6 unicast address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# next-hop-recursion
```

# no-dynamic-aging

Disables aging of ports that are dynamically assigned to the protocol or subnet-based VLANs.

## Syntax

**no-dynamic-aging**

**no no-dynamic-aging**

## Command Default

The dynamic protocol VLAN ages out after 10 or 20 minutes, if no packets are received.

## Modes

IP protocol VLAN configuration mode

IPX protocol VLAN configuration mode

AppleTalk protocol VLAN configuration mode

DECnet protocol VLAN configuration mode

NetBIOS protocol VLAN configuration mode

Other protocol VLAN configuration mode

IPV-6 protocol VLAN configuration mode

## Usage Guidelines

**NOTE**
Configure the command only if your configuration includes dynamically assigned VLAN memberships for protocol or subnet VLANs.

The **no** form of the command enables aging of the dynamic protocol VLAN.

## Examples

The following example shows how to configure dynamic aging.

```
device(config)# vlan 10 by port
device(config-vlan-10)# interface ethernet 1/1/1 to 1/1/5
device(config-vlan-10)# ip-proto name IP_Prot_VLAN
device(config-vlan-ip-proto)# no-dynamic-aging
```

# non-preempt-mode

Enables the non-preempt mode on all backups.

## Syntax

**non-preempt-mode**

**no non-preempt-mode**

## Command Default

By default, the non-preempt mode is disabled; preemption is enabled.

## Modes

VRID configuration mode

## Usage Guidelines

By default, a backup that has a higher priority than another backup that has become the master can preempt the master, and take over the role of master. If you want to prevent this behavior, disable preemption.

Preemption applies only to backups and takes effect only when the master has failed and a backup has assumed ownership of the VRID. The **non-preempt-mode** command prevents a backup with a higher priority from taking over as master from another backup that has a lower priority but has already become the master of the VRID.

Preemption is especially useful for preventing flapping in situations where there are multiple backups and a backup with a lower priority than another backup has assumed ownership, because the backup with the higher priority was unavailable when ownership changed.

If you enable the non-preempt mode (thus disabling the preemption feature) on all the backups, the backup that becomes the master following the disappearance of the master continues to be the master. The new master is not preempted.

The **no** form of the command disables the non-preempt mode.

## Examples

The following example enables the non-preemption mode.

```
device(config)# vlan 200
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# non-preempt-mode
```

# non-preempt-mode (VRRP)

Disables preempt mode for a Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRP-E) backup device.

## Syntax

**non-preempt-mode**

**no non-preempt-mode**

## Command Default

Preemption is enabled by default.

## Modes

VRID interface configuration mode

## Usage Guidelines

This command is supported in VRRP and VRRP-E. When the **non-preempt-mode** command is entered, a backup device with a higher VRRP priority is prevented from taking control of the virtual router ID (VRID) from another backup device that has a lower priority, but has already assumed control of the VRID. Disabling preemption is useful to prevent flapping when there are multiple backup devices and a backup with a lower priority assumes the role of master. When other backup devices with a higher priority are back online, the role of master can flap between devices.

In VRRP, the owner device always assumes the role of master when it comes back online, regardless of the preempt mode setting.

Enter **no non-preempt-mode** to re-enable preemption.

## Examples

The following example disables preempt mode for the virtual-router ID 1 session:

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.5.3/24
device(config-if-e1000-1/1/5)# ip vrrp vrid 1
device(config-if-e1000-1/1/5-vrid-1)# non-preempt-mode
```

# nonstop-routing (OSPF)

Enables nonstop routing (NSR) for OSPF.

## Syntax

**nonstop-routing**

**no nonstop-routing**

## Command Default

Enabled.

## Modes

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The **no** form of the command disables non-stop routing.

## Examples

The following example re-enables NSR on a device.

```
device# configuration terminal
device(config)# router ospf
device(config-ospf-router)# nonstop-routing
```

# ntp

Enables the Network Time Protocol (NTP) client and server mode.

## Syntax

**ntp**

**no ntp**

## Command Default

NTP services are disabled on all interfaces by default.

## Modes

Global configuration mode

## Usage Guidelines

Before you begin to configure NTP, you must use the **clock set** command to set the time on your device to within 1000 seconds of the Coordinated Universal Time (UTC).

The **no ntp** command disables NTP and removes the NTP configuration, including all static configuration as well as learned associations from NTP neighbors.

## Examples

The following example enables the NTP client and server mode.

```
device(config)# ntp
device(config-ntp)#
```

# ntp-interface

Enters Network Time Protocol (NTP) interface configuration mode.

## Syntax

**ntp-interface** { **management** *port* | **ethernet** *unit/slot/port* | **lag** *lag-id* | **ve** *ve-id* }

**no ntp-interface** { **management** *port* | **ethernet** *unit/slot/port* | **lag** *lag-id* | **ve** *ve-id* }

## Parameters

**management** *port*
Specifies the management interface.

**ethernet** *unit/slot/port*
Specifies the Ethernet interface.

**lag** *lag-id*
Specifies the LAG virtual interface.

**ve** *ve-id*
Specifies the Virtual Ethernet interface.

## Modes

NTP configuration mode

## Usage Guidelines

The broadcast server or client is configured on selected interfaces. To remove the NTP broadcast configurations on the specified interface, use the **no** form of this command.

The **no** form of the command returns to NTP configuration mode.

The **ntp-interface** command is a mode-change command.

## Examples

The following example enters the NTP interface configuration mode for Ethernet interface 1/1/1.

```
device(config)# ntp
device(config-ntp)# ntp-interface ethernet 1/1/1
device(config-ntp-if-e1000-1/1/1)#
```

The following example enters the NTP interface configuration mode for management interface 1.

```
device(config)# ntp
device(config-ntp)# ntp-interface management 1
device(config-ntp-mgmt-1)# exit
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include the LAG ID option. |

# Commands O, P, Q, R, and Sa through Si

## ocsp (PKI)

Sets the HTTP method for the Online Certificate Status Protocol (OCSP) request.

### Syntax

**ocsp** { **http post** }

**no ocsp** { **http post** }

### Command Default

By default, an HTTP "get" command is used for OCSP requests.

### Parameters

**http post**

Sets the method for OCSP requests.

### Modes

PKI trustpoint configuration sub-mode.

### Usage Guidelines

The no form of the command removes the configuration.

By default, an HTTP "get" command is used to reach the OCSP responder. The HTTP "get" method can be changed to an HTTP post using the command **ocsp http post**. The command is typically configured for the Linux operating system.

### Examples

The following example configures trustpoint abcd to use the HTTP post method for OCSP requests.

```
device# configure terminal
device(config)# pki trustpoint abcd
device(config-pki-trustpoint-abcd)# ocsp http post
device(config-pki-trustpoint-abcd)# revocation-check ocsp
device(config-pki-trustpoint-abcd)# ocsp-url http://15.1.1.1:2560
device(config-pki-trustpoint-abcd)# fingerprint 3C:EA:EC:E6:F1:DD:3B:86:65:DE:58:F4:A2:75:D8:63:6D:
23:68:40
device(config-pki-trustpoint-abcd)# exit
device(config)#
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# ocsp-url (PKI)

Defines the URL to be used for Online Certificate Status Protocol (OCSP) requests.

## Syntax

**ocsp-url** { *url* }

**no oscp-url** { *url* }

## Command Default

## Parameters

*url*

Configures the URL for OCSP requests.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

## Examples

The following example sets the url for OCSP requests to http://15.1.1.1:2560.

```
device# configure terminal
device(config)# pki trustpoint abcd
device(config-pki-trustpoint-abcd)# ocsp http post
device(config-pki-trustpoint-abcd)# revocation-check ocsp
device(config-pki-trustpoint-abcd)# ocsp-url http://15.1.1.1:2560
device(config-pki-trustpoint-abcd)# fingerprint 3C:EA:EC:E6:F1:DD:3B:86:65:DE:58:F4:A2:75:D8:63:6D:
23:68:40
device(config-pki-trustpoint-abcd)# exit
device(config)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# openflow enable

Enables or disables the OpenFlow hybrid port-mode on the port.

## Syntax

**openflow enable** [ **layer2** | **layer3** | **layer23** [**hybrid-mode** ] ]

**no openflow enable** [ **layer2** | **layer3** | **layer23** [**hybrid-mode** ] ]

## Parameters

**layer2**
Enables Layer 2 matching mode for flows.

**layer3**
Enables Layer 3 matching mode for flows.

**layer23 hybrid-mode**
Enables Layer 2 and Layer 3 matching mode for flows with an option for hybrid port-mode.

## Modes

Global configuration mode.

Interface configuration mode.

## Usage Guidelines

In interface configuration mode, this command enables Layer 2 or Layer 3 matching mode for flows with an optional enabling of hybrid port-mode.

**NOTE**
OpenFlow must be globally enabled before the Layer 2 or Layer 3 matching modes can be specified.

## Examples

After OpenFlow 1.3 is enabled, the following example configures Layer 2 and Layer 3 matching mode for flows.

```
device# configure terminal
device(config)# openflow enable ofv130
device(config)# interface ethernet 1/1/1
device(config-if-1/1/1)# openflow enable layer 23
```

## History

| Release | Command History |
|---|---|
| 8.0.20 | This command was introduced. |

# openflow purge-time

Configures the maximum amount of time (in seconds) before stale flows are purged from the OpenFlow flow table after a switchover, failover, or OS upgrade.

## Syntax

**openflow purge-time seconds**

**no openflow purge-time seconds**

## Command Default

The value of the OpenFlow purge timer is the default value for normal circumstances.

## Parameters

**seconds**

Specifies the maximum amount of time (in seconds), before stale flows are purged. The range is from 1 through 600. The default is 240 seconds.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

You can configure a larger value for the OpenFlow purge timer, if delay is anticipated in learning the flows from controller after switch-over.

## Examples

The following example sets the OpenFlow purge time to 500 seconds:

```
device(config)# openflow purge-time 500
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# optical-monitor

Configures the device to monitor optical transceivers in the system.

## Syntax

**optical-monitor** [ *alarm-interval* ]

**no optical-monitor** [ *alarm-interval* ]

## Command Default

The default and minimum timer for the ICX 7450 and ICX 7750 devices is 8 minutes.

The default timer for the ICX 7250 device is 3 minutes but can be set as low as 1 minute.

## Parameters

*alarm-interval*

Specifies the interval between at which alarms and warning messages are sent.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

You can configure your Ruckus device to monitor optical transceivers in the system. When Digital Optical Monitoring (DOM) is enabled, the system monitors the temperature and signal power levels for the optical transceivers. Console messages and syslog messages are sent when optical operating conditions fall below or rise above the QSFP+, SFP, or SFP+ manufacturer-recommended thresholds.

The commands **no optical-monitor** and **optical-monitor 0** perform the same function; that is, they both disable DOM.

## Examples

The following example enables optical monitoring on all Ruckus-qualified optics installed in the device.

```
device(config)# optical-monitor
```

The following example enables optical monitoring on a specific port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# optical-monitor
```

The following example sets the alarm interval to 10 minutes.

```
device(config)# interface ethernet 1/1/1 to 1/1/4
device(config-mif-e10000-1/1/1-1/1/4)# optical-monitor 10
```

# option

Specifies Dynamic Host Configuration Protocol (DHCP) options to be exchanged between the server and the client.

## Syntax

**option** *option-number* { **ascii** | **hex** | **ip** } *option-value*

## Parameters

*option-number*
: Specifies the DHCP generic option number to be exchanged between server and client.

**ascii**
: Specifies the *option-value* is an ASCII string (up to 128 characters).

**hex**
: Specifies the *option-value* is a hexadecimal value (64-byte hex sequence, up to 128 characters).

**ip**
: Specifies the *option-value* is an IP address. You can configure up to a maximum of three IP addresses separated by a space.

*option-value*
: The value of the option; for example, an IP address or vendor-specific information.

## Modes

DHCP server pool configuration mode

## Usage Guidelines

This command is used to specify DHCP options that the DHCP server passes to clients.

For example, you can configure option 3, so the DHCP server passes the IP address of the default routers to the client.

Only one default router can be specified. Do not enter multiple router addresses.

Another example is option 43, allowing the DHCP server to pass vendor-specific information, in the form of a hex string or an ASCII string, to the clients that receive the DHCP ACK. With this example, configuring DHCP option 60 helps in identifying the incoming DHCP client. If the vendor class identifier (VCI) advertised by the DHCP client matches with the DHCP server, the server makes a decision to exchange the vendor-specific information configured as part of DHCP option 43.

## Examples

The following example configures option 3, using the IP address format, to specify the default router available to the client.

```
device# configure terminal
device(config)# ip dhcp-server pool ruckus
device(ip dhcp-server pool ruckus)# option 3 ip 10.10.10.1
```

The following example configures option 12, using the ASCII format, to specify the hostname server available to the client.

```
device# configure terminal
device(config)# ip dhcp-server pool ruckus
device(ip dhcp-server pool ruckus)# option 12 ascii myhostname
```

The following example configures option 43, using the hex option for comma-separated Ruckus AP IP addresses configuration.

```
device# configure terminal
device(config)# ip dhcp-server pool ruckus
device(ip dhcp-server pool ruckus)# option 43 hex 0x061731302e31302e31302e31302c31322e31322e31322e3132
```

The following example configures option 67, specifying both the image type and flash location.

```
device# configure terminal
device(config)# ip dhcp-server pool ruckus
device(ip dhcp-server pool ruckus)# option 67 ascii "fi8080_manifest.txt router primary"
```

# History

| Release version | Command history |
|---|---|
| 08.0.30mb | This command was introduced. |
| 08.0.70 | This command was expanded to support DHCP generic server options. |
| 08.0.80 | This command was modified to specify both the image type and flash location for option 67.. |

# originator-id

Configures MSDP to use the specified interface IP address as the IP address of the rendezvous point (RP) in a source-active (SA) message.

## Syntax

**originator-id** *type number*

**no originator-id** *type number*

## Command Default

MSDP uses the IP address of the originating RP in the RP address field of the SA message.

## Parameters

*type*

Specifies the type of interface used by the RP. You can use Ethernet, loopback, and virtual routing interfaces (ve).

*number*

Specifies the interface number. For example, the Ethernet port number, loopback number, or virtual routing interface number.

## Modes

MSDP router configuration mode

MSDP router VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the default

## Examples

This example configures an interface IP address to be the IP address of the RP.

```
Device(config)# interface loopback 2
Device(config-lbif-2)# ip address 2.2.1.99/32
Device(config)# router msdp
Device(config-msdp-router)# originator-id loopback 2
Device(config-msdp-router)# exit
```

This example configures an interface IP address to be the IP address of the RP on a VRF named blue.

```
Device(config)# interface loopback 2
Device(config-lbif-2)# ip address 2.2.1.99/32
Device(config)# router msdp vrf blue
Device(config-msdp-router-vrf blue)# originator-id loopback 2
Device(config-msdp-router-vrf blue)# exit
```

# other-proto

Configures the other protocol VLAN and enters the other protocol VLAN configuration mode.

## Syntax

**other-proto** [ **name** *string* ]

**no other-proto** [ **name** *string* ]

## Command Default

IP protocol VLANs are configured.

## Parameters

**name** *string*

Specifies the name of the other protocol VLAN configuration. The name can be up to 32 characters in length.

## Modes

VLAN configuration mode

IP protocol VLAN configuration mode

IPX protocol VLAN configuration mode

IPv6 protocol VLAN configuration mode

DECnet protocol VLAN configuration mode

NetBIOS protocol VLAN configuration mode

AppleTalk protocol VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the other protocol VLANs.

## Examples

The following example shows how to configure the other protocol VLAN.

```
device(config)# ipx-proto name Brown
device(config-vlan-ipx-proto)# other-proto name Block_other_proto
device(config-vlan-other-proto)# no dynamic
```

# overlay-gateway

Configures an overlay-gateway name and enters gateway configuration mode.

## Syntax

**overlay-gateway** *gateway-name*

**no overlay-gateway** *gateway-name*

## Command Default

Overlay gateway name is not configured.

## Parameters

*gateway-name*

Specifies the name of an overlay-gateway interface. Maximum length is 64 characters.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command removes the gateway configuration.

The command is supported only on ICX 7750 devices.

Only one overlay-gateway is supported.

## Examples

The following example configures the name gate1 for the overlay-gateway, and places the device in overlay-gateway configuration mode.

```
device(config)#overlay-gateway gate1
device(config-overlay-gw-gate1)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# owner

Designates a virtual router as the Virtual Router Redundancy Protocol (VRRP) owner and configures priority and track values.

## Syntax

**owner** [ **priority** *value* ] [ **track-priority** *value* ]

**no owner** [ **priority** *value* ] [ **track-priority** *value* ]

## Command Default

No virtual routers are designated as the VRRP owner.

## Parameters

**priority** *value*

Abdicates owner status by setting a value that is lower than the backup default priority value. Value can be from 1 to 254. Default is 100.

**track-priority** *value*

Sets the priority value if the tracked port fails. Value can be from 1 to 254. Default is 2.

## Modes

VRID interface configuration mode

## Usage Guidelines

This command specifies that the device on which it is configured owns the IP address that is associated with the virtual router; making this device the default VRRP master router with its priority set to 255.

This command must be entered before the **ip-address** command can be configured for a VRRP virtual router ID (VRID).

The **no** form of this command removes the virtual router configuration.

## Examples

The following example configures the device as the VRRP owner.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.5.1
device(config-if-e1000-1/1/6-vrid-1)# activate
```

The following example configures the device as the VRRP owner and sets the track priority to 10.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# owner track-priority 10
device(config-if-e1000-1/1/6-vrid-1)# ip-address 10.53.5.1
device(config-if-e1000-1/1/6-vrid-1)# activate
```

# password

Specifies the password to be used for the key in encrypted form for the cryptographic algorithm.

## Syntax

**password** *passphrase*

**no password** *passphrase*

## Parameters

*passphrase*

Specifies the password for the key for the cryptographic algorithm, which is encrypted.

## Modes

Key ID configuration mode

## Usage Guidelines

A key is considered valid only if the key has not expired, and the password and authentication algorithm have been specified.

The **no** form of the command removes the password from the key.

## Examples

The following example specifies the password for the key for the cryptographic algorithm.

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)# key-id 10
device(config-keychain-xprotocol-key-10)# password pass
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# packet-inerror-detect

Enables the monitoring of a port for inError packets and defines the maximum number of inError packets allowed for the port during the configured sampling interval.

## Syntax

**packet-inerror-detect** *inError-count*

**no packet-inerror-detect** *inError-count*

## Command Default

The Packet InError Detect feature is disabled for the port.

## Parameters

*inError-count*

Specifies the maximum number of inError packets that are allowed for a port during the configured sampling interval. The value can range from 10 through 4294967295.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command disable monitoring of inError packets for the port.

If the number of inError packets received at a port exceeds the default value for two consecutive sampling windows, the port is set to the error-disabled state.

**NOTE**
To enable monitoring of inError packets for the port only, you must first use the **errdisable packet-inerror-detect** command in global configuration mode to globally enable monitoring for inError packets on the device.

## Examples

The following example displays the maximum number of allowed inError packets for a port set to the value 10.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# packet-inerror-detect 10
```

## History

| Release version | Command history |
|---|---|
| 07.3.00g | This command was introduced. |

# pass-through

Enables pass-through which allows certain protocol packets to pass through ports that are enabled for Flexible authentication.

## Syntax

**pass-through** { **cdp** | **fdp** | **lldp** }

**no pass-through**{ **cdp** | **fdp** | **lldp** }

## Command Default

Pass-through is not enabled.

## Parameters

**cdp**

Specifies the Cisco Discovery Protocol to pass through.

**fdp**

Specifies the Foundry Discovery Protocol to pass through.

**lldp**

Specifies the Link Layer Discovery Protocol to pass through.

## Modes

Authentication mode

## Usage Guidelines

This command specifies the protocols to be passed through even though the client is not authenticated.

The **no** form of the command disables pass-through.

## Examples

The example enables LLDP for pass-through.

```
device(config)# authentication
device(config-authen)# pass-through lldp
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# pdu-rate (EFM-OAM)

Configures the number of Protocol Data Units (PDUs) to be transmitted per second by the Data Terminal Equipment (DTE).

## Syntax

**pdu-rate** *value*

**no pdu-rate** *value*

## Command Default

The default value is one PDU per second.

## Parameters

*value*

Specifies the number of PDUs to be sent per second. The value range can be from 1 through 10 PDUs per second.

## Modes

EFM-OAM protocol configuration mode

## Usage Guidelines

If the PDU rate is configured as 10 packets per second, PDUs may not get transmitted in a timely manner according to the configured PDU rate.

The **no** form of the command restores the default value of one PDU per second.

## Examples

The following example configures the PDU rate as 6 PDUs per second.

```
device(config)# link-oam
device(config-link-oam)# pdu-rate 6
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# pe-id

Assigns and reserves an ID for an SPX port extender (PE) unit.

## Syntax

**pe-id** *port unit-id* [ *port* | *name* ]

**no pe-id** *port unit-id* [ *port* | *name* ]

**pe-id** *name unit-id1* [ *unit-id2 unit-id3 unit-id4* ] [ *port* | *name* ]

**no pe-id** *name unit-id1* [ *unit-id2 unit-id3 unit-id4* ] [ *port* | *name* ]

## Command Default

By default, the system generates the PE IDs for attached PE units.

## Parameters

*port*

In the form *unit/slot/port*, designates the CB SPX port or a port in the CB SPX LAG that links to the PE *unit-id*.

*name*

The PE group name for an SPX port or SPX LAG that is associated with the ID or IDs that follow. Up to four IDs may be associated with the PE group name. The PE group name must also be defined in CB configuration mode.

*unit-id*

Designates the PE ID or IDs associated with the port or the PE group name. PE ID values range from 17 through 56.

## Modes

CB configuration mode

## Usage Guidelines

The **no** form of the command removes the PE ID and any associated configuration.

The second set of [ *port* | *name* ] parameters after the *unit-id*s provides an option to specify a ring topology. (Although a ring topology is not supported in FastIron 8.0.40, the configuration is allowed for compatibility with future releases.)

The output of the **show running-config** command shows the merged result of system-generated PE IDs and the user's reserved PE ID configuration. The system overwrites user entries if there is a conflict.

If a reserved stack unit is removed, its associated SPX port configuration and PE ID configuration are removed.

Users are allowed to change the PE ID configuration of live PE units as long as the new configuration does not alter the topology of the live units.

# Examples

The following example assigns the PE ID 20 to the first PE that attaches to SPX port 2/1/15. The next PE unit that links to PE unit 20 is assigned PE ID 22.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# spx-port 2/1/15 pe-group bld1-f13-stk2
device(config-spx-cb)# pe-id bld1-f13-stk2 20 22
```

The PE group name or any port in a LAG can be used to reserve the PE ID. The following three examples configure the same PE ID, assuming both ports 1/1/10 and 1/1/11 are in the same SPX LAG and have the PE group name shown.

```
device(config)# pe-id 1/1/10 20

device(config)# pe-id 1/1/11 20

device(config)# pe-id bld1-f13-stk7 20
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# pe-name

Assigns a name to the PE unit.

## Syntax

**pe-name** *string*

**no pe-name**

## Command Default

By default, no name is assigned to the PE unit.

## Parameters

*string*

Character string that specifies the name for this PE unit.

## Modes

PE configuration mode

Provisional-PE configuration mode

## Usage Guidelines

The **no** form of the command removes the name assigned to the PE (no string match is required).

The PE name must be unique within the SPX domain.

The PE name is an identifier only; it cannot be used as a replacement, for example, for a port number, to change other configuration.

## Examples

In the following example, a CB gives a name to PE unit 18.

```
device# configure terminal
device(config)# spx unit 18
device(config-spx-unit-18)# pe-name bldg2-floor2-stk 18
device(config-spx-unit-18)# exit
device(config)# exit
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# peer

Configures the software clock to synchronize a peer or to be synchronized by a peer.

## Syntax

**peer** { *ipv4-address* | *ipv6-address* } [ **key** *key-id* ] [ **maxpoll** *interval* ] [ **minpoll** *interval* ] [ **version** *version-number* ] [ **burst** ]

**no peer** { *ipv4-address* | *ipv6-address* } [ **key** *key-id* ] [ **maxpoll** *interval* ] [ **minpoll** *interval* ] [ **version** *version-number* ] [ **burst** ]

## Command Default

A peer is not configured.

## Parameters

*ipv4-address*
: Specifies the IPv4 address of the peer providing the clock synchronization.

*ipv6-address*
: Specifies the IPv6 address of the peer providing the clock synchronization.

**key** *key-id*
: Specifies the authentication key. The value can range from 1 through 65535. By default, no authentication key is configured.

**maxpoll** *interval*
: Specifies the longest polling interval. The range is from 4 through 17. The default is 10. The interval argument is a power of 2 (4=16, 5=32, 6=64, 7=128, 8=256, 9=512, and so on).

**minpoll** *interval*
: Specifies the shortest polling interval. The range is from 4 through 17. The default is 6. The interval argument is a power of 2 (4=16, 5=32, 6=64, 7=128, 8=256, 9=512, and so on).

**version** *version-number*
: Specifies the Network Time Protocol (NTP) version number. Valid values are 3 and 4. The default value is 4.

**burst**
: Sends a burst of packets to the server at each polling interval.

## Modes

NTP configuration mode

## Usage Guidelines

NTP peer mode is intended for configurations where a group of devices operate as mutual backups for each other. If one of the devices loses a reference source, the time values flow from the surviving peers to all the others.

A maximum of eight NTP peers can be configured.

**NOTE**
The **peer** command is not effective if NTP is enabled in client-only mode.

**NOTE**
If the peer is a member of a symmetric passive association, configuring the **peer** command will fail.

The **no** form of the command disables the software clock to synchronize a peer.

# Examples

The following example configures the software clock.

```
device(config)# ntp
device(config-ntp)# peer 10.2.2.2 key 23 maxpoll 15 minpoll 7 version 4 burst
```

# peer disable-fast-failover

Disables the MCT fast-failover mode.

## Syntax

**peer** *peer-ip* **disable-fast-failover**

**no peer** *peer-ip* **disable-fast-failover**

## Command Default

Fast-failover is configured on the device.

## Parameters

*peer-ip*

Specifies the IP address of the peer device.

## Modes

Cluster configuration mode

## Usage Guidelines

The following failover modes can be configured with MCT:

- Fast-failover (default) - As soon as the ICL interface goes down, the MCT control path between the two peer devices goes down. All the remote MAC addresses are flushed.
- Slow-failover - Even if the ICL interface goes down, the CCP waits for the hold-time before taking the MCT control path between the two peer devices down. Remote MAC addresses are flushed only when the MCT control path between the two peer devices is down.

The **no** form of the command re-enables fast-failover.

## Examples

The following example shows how to disable fast-failover.

```
device(config)# cluster SX
device(config-cluster-SX)# peer 10.1.1.3 disable-fast-failover
```

# peer timers

Configures the keep-alive and hold-time timers for peer devices.

## Syntax

**peer** *peer-ip* **timers keep-alive** *keep-alive-timer* **hold-time** *hold-timer*

**no peer** *peer-ip* **timers keep-alive** *keep-alive-timer* **hold-time** *hold-timer*

## Command Default

The default value for the keep-alive timer is 10 seconds.

The default value for the hold-time timer is 90 seconds.

## Parameters

*peer-ip*
Specifies the IP address of the cluster peer.

**keep-alive** *keep-alive-timer*
Specifies the keep-alive interval in seconds. The value can range from 0 through 21845 seconds.

**hold-time** *hold-timer*
Specifies the hold-time interval in seconds. The value can range from 3 through 65535 seconds (or 0 if the keep-alive timer is set to 0).

## Modes

Cluster configuration mode

## Usage Guidelines

The *peer-ip* parameter should be in the same subnet as the cluster management interface. The hold-time must be at least three times the keep-alive time.

**NOTE**
The keep-alive VLAN and keep-alive timers are not related. The keep-alive timer is used by CCP.

The **no** form of the command sets the timers to the default values.

## Examples

The following example shows how to configure the peer timers.

```
device(config)# cluster SX 400
device(config-cluster-SX))# peer 10.1.1.3 timers keep-alive 40 hold-time 120
```

# peer-info

Configures the peer system ID and system key for a single dynamic Link Aggregation Group (LAG).

## Syntax

**peer-info sys-mac** *mac-address* **sys-pri** *number* **key** *key number*

**no peer-info sys-mac** *mac-address* **sys-pri** *number* **key** *key number*

## Command Default

The peer information of any one of the ports of a dynamic LAG that forms the first LACP trunk within that dynamic LAG, is considered as the peer information.

## Parameters

**sys-mac** *mac-address*
Specifies the system's peer Ethernet MAC address.

**sys-pri** *number*
Specifies the LACP system priority for the system's peer. Valid numbers range from 0 through 65535.

**key** *key number*
Specifies the LACP key value. Valid key numbers range from 1 through 65535.

## Modes

LAG configuration mode

## Usage Guidelines

The **no** form of the command removes the peer information configuration for the dynamic LAG.

## Examples

The following example configures the peer system with a system priority of 10 and an LACP key value of 10000.

```
device(config)# lag R4-dyn2
device(config-lag-R4-dyn2)# peer-info sys-mac 0000.0000.0003 sys-pri 10 key 10000
```

## History

| Release version | Command history |
|---|---|
| 8.0.30d | This command was introduced. |

# permit (extended IPv4 ACLs)

Inserts filtering rules in IPv4 extended named or numbered ACLs that will permit packets.

## Syntax

Use the following syntax to define a TCP or UDP rule that will permit packets:

[ **no** ] **permit** { **tcp** | **udp** } { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ *source-comparison-operators* ] { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ **established** ] [ *destination-comparison-operators* ] [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define an ICMP rule that will permit packets:

[ **no** ] **permit icmp** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ *icmp-num* | *icmp-type* ] [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define a rule for protocols other than TCP, UDP, or ICMP that will permit packets:

[ **no** ] **permit** *ip-protocol* { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

**no sequence** *seq-num*

## Parameters

*ip-protocol*

Specifies the type of IPv4 packet to filter. You can either specify a protocol number (from 0 through 255) or a supported protocol name. For a complete list of protocols, type **?** after **permit**. Supported protocols include:

- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **igrp**—Internet Gateway Routing Protocol
- **ip**—any IPv4 protocol
- **ospf**—Open Shortest Path First
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

*S_IPaddress*

Specifies a source address for which you want to filter the subnet.

*mask*

Defines a mask, whose effect is to specify a subnet that includes the source address that you specified. For options to specify the mask, see the Usage Guidelines.

**host**

Specifies the source as a host.

*S_IPaddress*

Specifies the source address of the host.

**any**

Specifies all source addresses.

*source-comparison-operators* and *destination-comparison-operators*

If you specified **tcp** or **udp**, the following optional operators are available:

**eq**

Specifies the address is equal to the port name or number you enter after **eq**.

**gt**

Specifies port numbers that are equal to or greater than the port number or that are equal to or greater than the numeric equivalent of the port name you enter after **gt**.

**lt**

Specifies port numbers that are equal to or less than the port number or that are equal to or less than the numeric equivalent of the port name you enter after **lt**.

**neq**

Specifies all port numbers except the port number or port name you enter after **neq**.

**range**

Specifies all port numbers that are between the first port name or number and the second name or number you enter following the **range** keyword. Enter the range as two values separated by a space. The first port number in the range must be less than the last number in the range. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: 23 53 .

*D_IPaddress*

Specifies a destination address for which you want to filter the subnet.

*mask*

Defines a subnet mask that includes the destination address that you specified. For mask options, refer to the Usage Guidelines.

**host**

Specifies a host as destination.

*D_IPaddress*

Specifies the destination address of the host.

**any**

Specifies all destination addresses.

**established**

(For TCP rules only) Filter packets that have the Acknowledgment (ACK) or Reset (RST) flag set. This policy applies only to established TCP sessions, not to new sessions.

*icmp-num* | *icmp-type*

(For ICMP only) Specifies a named or numbered message type.

*icmp-num*

Specifies a numbered message type. Use this format if the rule also needs to include **precedence**, **tos** , one of the DSCP options, one of the 802.1p options, **internal-priority-marking** , or **traffic-policy**.

**any-icmp-type**
Specifies any ICMP type.

**echo**
Specifies an echo request (ping).

**echo-reply**
Specifies an echo reply.

**information-request**
Specifies an information request.

**mask-reply**
Specifies an address mask reply.

**mask-request**
Specifies an address mask request.

**parameter-problem**
Specifies a parameter problem.

**redirect**
Specifies a redirect message.

**source-quench**
Specifies a relieve congestion message.

**time-exceeded**
Specifies a time exceeded message.

**timestamp-reply**
Specifies a timestamp reply.

**timestamp-request**
Specifies a timestamp request.

**unreachable**
Specifies a destination-unreachable message.

**precedence** { *precedence-name* | *precedence-value* }
Specifies a *precedence-name* or corresponding *precedence-value*, as follows:

**0** or **routine**
Specifies routine precedence.

**1** or **priority**
Specifies priority precedence.

**2** or **immediate**
Specifies immediate precedence.

**3** or **flash**
Specifies flash precedence.

**4** or **flash-override**
Specifies flash-override precedence.

**5** or **critical**
Specifies critical precedence.

**6** or **internet**
Specifies internetwork control precedence.

**7** or **network**
Specifies network control precedence.

**tos** { *tos-name* | *tos-value* }
Specifies a type of service (ToS). Enter either a supported *tos-name* or the equivalent *tos-value*.

**0** or **normal**
Specifies normal ToS.

**1** or **min-monetary-cost**
Specifies min monetary cost ToS.

**2** or **max-reliability**
Specifies max reliability ToS.

**4** or **max-throughput**
Specifies max throughput ToS.

**8** or **min-delay**
Specifies min-delay ToS.

**dscp-matching** *dscp-value*
Filters by DSCP value. Values range from 0 through 63.

**dscp-marking** *dscp-value*
Assigns the DSCP value that you specify to the packet. Values range from 0 through 63.

**802.1p-priority-matching** *802.1p-value*
Filters by 802.1p priority, for rate limiting. Values range from 0 through 7.

**802.1p-priority-marking** *802.1p-value*
Assigns the 802.1p value that you specify to the packet. Values range from 0 through 7.

**internal-priority-marking** *queuing-priority*
Assigns the internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**802.1p-and-internal-marking** *priority-value*
Assigns the identical 802.1p value and internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**traffic-policy** *name*
Enables the device to limit the rate of inbound traffic and to count the packets and bytes per packet to which ACL permit clauses are applied. For configuration procedures and examples, refer to the chapter "Traffic Policies" in the *Ruckus FastIron Traffic Management Configuration Guide*.

**log**
Enables SNMP traps and Syslog messages for the rule. In addition, logging must be enabled using the **acl-logging** command.

**mirror**
Mirrors packets matching the rule.

# Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

# Usage Guidelines

Extended ACLs permit traffic according to source and destination addresses, port protocol, and other IPv4 frame content. You can also enable logging and mirroring.

The order of the rules in an ACL is critical, as the first matching rule stops further processing.

You can specify a mask in either of the following ways:

- Wildcard mask format (for example, 0.0.0.255). The advantage of this format is that it enables you mask any bit, for example by specifying 0.255.0.255.
- Classless Interdomain Routing (CIDR) format, in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 in the wildcard mask format.

If you specify **icmp** and also specify the **any-icmp-type** option, the following QoS options are not available: **dscp-marking**, **dscp-matching**, **internal-priority-marking**, **802.1p-priority-marking**, ond **802.1p-priority-matching**.

On the Ruckus ICX 7150 and Ruckus ICX 7750, ACL logging is not supported for egress ACLs.

When specifying type of service (ToS), you can indicate multiple *tos-value* options by entering the sum of the needed ToS options. For example, to specify both **max-reliability** and **min-delay**, enter **10**. To specify all options, enter **15**. Values range from **0** through **15**.

In a rule that includes one or more of the following parameters, the **log** keyword is ignored:

- **dscp-matching**
- **dscp-marking**
- **802.1p-priority-matching**
- **802.1p-priority-marking**
- **802.1p-and-internal-marking**

For details on 802.1p priority matching, refer to "Inspecting the 802.1p bit in the ACL for adaptive rate limiting" in the *Ruckus FastIron Traffic Management Configuration Guide*.

To delete a permit rule from an ACL, type **no** followed by the full command syntax.

# Examples

The following ACL, applied to an Ethernet interface, creates an extended ACL that permits any source or destination address for any IPv4 protocol.

```
device# configure terminal
device(config)# ip access-list extended "block Telnet"
device(config-ext-nacl)# permit ip any any
device(config-ext-nacl)# exit
device(config)# interface ethernet 1/1/1
device(config-if-1/1/1)# ip access-group "block Telnet" in
```

# permit (standard IPv4 ACLs)

Inserts filtering rules in IPv4 standard named or numbered ACLs that will permit packets.

## Syntax

**permit** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

**no permit** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

## Parameters

*S_IPaddress*
: Specifies a source address for which you want to filter the subnet.

*mask*
: Defines a subnet mask that includes the source address you specified.

**host**
: Indicates the source IP address is a host address.

*S_IPaddress*
: Specifies source address.

**any**
: Specifies all source addresses.

**log**
: Enables logging for the rule.

**mirror**
: Mirrors packets matching the rule.

## Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

## Usage Guidelines

This command configures rules to permit traffic based on source addresses. You can also enable logging and mirroring.

Standard ACLs permit traffic according to source address only.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list. Such a rule is automatically assigned the next multiple of 10 as a sequence number.

You can specify a mask in either of the following ways:

- Wildcard mask format. The advantage of this format is that it enables you to mask any bit, for example by specifying 0.255.0.255.

- Classless Interdomain Routing (CIDR) format—in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 in the wildcard mask format.

On Ruckus ICX 7150 and Ruckus ICX 7750 devices, ACL logging is not supported for egress ACLs.

For the **log** keyword to trigger a log entry, logging must be enabled with the **acl-logging** command.

To delete a rule from an ACL, use the **no permit** command followed by the full command syntax.

# Examples

The following example shows how to configure a standard numbered ACL and apply it to incoming traffic on port 1/1/1.

```
device# configure terminal
device(config)# ip access-list standard 1
device(config-std-nacl)# deny host 10.157.22.26 log
device(config-std-nacl)# deny 10.157.29.12 log
device(config-std-nacl)# deny host IPHost1 log
device(config-std-nacl)# permit any
device(config-std-nacl)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip access-group 1 in
```

# phy cable diagnostics tdr

Runs the VCT TDR test on the specified port.

## Syntax

**phy cable-diagnostics tdr** *stackid/slot/port*

## Parameters

*stackid/slot/port*

Specifies the interface (port), by device, slot, and port number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to clear TDR test registers before every TDR cable diagnostic test.

Before executing this command, use the **clear cable-diagnostics tdr** command to clear any previous TDR test results.

Display diagnostic test results using the **show cable-diagnostics tdr stackid/slot/port** command.

## Examples

The following example clears test registers for the interface and then runs the TDR diagnostic test for port 3 on slot 2 of the first device in the stack.

```
device# clear cable-diagnostics tdr 1/2/3
device# phy cable-diag tdr 1/2/3
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced for ICX 6610, ICX 6430, ICX 6430-C, ICX 6450, and ICX6450-C devices. |

# phy-fifo-depth

Configures the depth of the transmit and receive FIFOs.

## Syntax

**phy-fifo-depth** *value*

**no phy-fifo-depth** *value*

## Command Default

The default value is 0.

## Parameters

*value*

Specifies the setting value. There are 4 settings 0 through 3 with 0 as the default.

## Modes

Interface configuration mode

## Usage Guidelines

PHY devices on Ruckus devices contain transmit and receive synchronizing FIFOs to adjust for frequency differences between clocks. The **phy-fifo-depth** command allows you to configure the depth of the transmit and receive FIFOs. A higher setting indicates a deeper FIFO.

The default setting works for most connections. However, if the clock differences are greater than the default can handle, CRCs and errors will begin to appear on the ports. Raising the FIFO depth setting adjusts for clock differences.

It is recommend that you disable the port before applying this command, and then re-enable the port. Applying the command while traffic is flowing through the port can cause CRC and other errors for any packets that are passing through the PHY while the command is being applied.

This command can be issued for a single port from interface configuration mode or for multiple ports from MIF configuration mode.

The **no** form of the command removes the depth of the transmit.

## Examples

The following example configures the FIFO depth for a single port.

```
device(config)# interface ethernet 1/1/21
device(config-if-e1000-1/1/21)# phy-fifo-depth 2
```

The following example configures the FIFO depth for multiple ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/5
device(config-mif-1/1/1-1/1/5)# phy-fifo-depth 1
```

# ping

Verifies whether a device can reach another device through the network.

## Syntax

**ping** { *ip-addr* | *host-name* | **vrf** *vrf-name* | **ipv6** [ *ipv6-addr* | *host-name* | **vrf** *vrf-name* ] [ **outgoing-interface** *type number* ] } [ **source** *ip-addr* ] [ **count** *num* ] [ **timeout** *msec* ] [ **ttl** *num* ] [ **size** *num* ] [ **quiet** ] [ **numeric** ] [ **no-fragment** ] [ **verify** ] [ **data** *1-to-4-byte-hex* ] [ **brief** [ **max-print-per-sec** *number* ] ]

## Parameters

*ip-addr*
: Specifies the IP address of the device to be pinged.

*host-name*
: Specifies the host name of the device to be pinged.

**vrf** *vrf-name*
: Specifies the Virtual Routing and Forwarding (VRF) instance of the device to be pinged.

**ipv6** *ipv6-addr*
: Specifies the IPv6 address, host name or VRF instance of the device to be pinged.

**outgoing-interface** *type number*
: Specifies an interface over which to verify connectivity.

**source** *ip-addr*
: Specifies an IP address to be used as the origin of the ping packets.

**count** *num*
: Specifies the number of ping packets that the device sends. The value can range from 1 to 4294967296. The default is 1.

**timeout** *msec*
: Specifies the time, in milliseconds for which the device waits for a reply from the pinged device. The value can range from 1 to 4294967296. The default is 5000 (5 seconds).

**ttl** *num*
: Specifies the time to live as a maximum number of hops. The value can range from 1 to 255. The default is 64.

**size** *num*
: Specifies the size of the ICMP data portion of the packet, in bytes. This is the payload and does not include the header. The value can range from 0 to 10000. The default is 16.

**no-fragment**
: Turns on the "don't fragment" bit in the IP header of the ping packet. This option is disabled by default.

**quiet**
: Hides informational messages such as a summary of the ping parameters sent to the device and instead only displays messages indicating the success or failure of the ping. This option is disabled by default.

**verify**
: Verifies that the data in the echo packet (the reply packet) is the same as the data in the echo request (the ping). By default the device does not verify the data.

**data***1-to-4-byte-hex*

Specifies a data pattern for the payload instead of the default data pattern, "abcd", in the packet data payload. The pattern repeats itself throughout the ICMP message (payload) portion of the packet.

**brief**

Specifies that the ping test characters are to be displayed. For more information, refer to the Usage Guidelines section.

**max-print-per-sec** *number*

Specifies the maximum number of target responses that the device can display per second while in brief mode. The value can range from 0 to 2047. The default is 511.

# Modes

All configuration modes

# Usage Guidelines

The following ping test characters are supported:

- !—Indicates that a reply was received.
- .—Indicates that the network server timed out while waiting for a reply.
- U—Indicates that a destination unreachable error PDU was received.
- I—Indicates that the user interrupted the ping.

For numeric parameter values, the command does not check that the value you enter is within the allowed range. Instead, if you do exceed the range for a numeric value, the software rounds the value to the nearest valid value.

**NOTE**

If the device is a Layer 2 switch or Layer 3 switch, you can use the host name only if you have already enabled the Domain Name Server (DNS) resolver feature on the device from which you are sending the ping.

# Examples

The following example checks the connectivity to the device at IP address 10.31.248.12.

```
device> ping 10.31.248.12
Sending 1, 16-byte ICMP Echo to 10.31.248.12, timeout 5000 msec, TTL 64
Type Control-c to abort
Reply from 10.31.248.12    : bytes=16 time=33ms TTL=63
Success rate is 100 percent (1/1), round-trip min/avg/max=33/33/33 ms.
```

# pki authenticate

Authenticates the certificate authority (CA) to the router by obtaining the self-signed certificate of the CA.

## Syntax

**pki authenticate** { *trustpoint* }

**no pki authenticate** { *trustpoint* }

## Command Default

The CA is not authenticated.

## Parameters

*trustpoint*

Specifies the name of the trustpoint (CA) to be authenticated.

## Modes

Global configuration mode.

## Usage Guidelines

The no form of the command removes both authentication and enrollment of the trustpoint.

The certificate obtained from the CA is saved to the router.

The self-signed certificate obtained from the CA contains the public key for the CA.

## Examples

The following example authenticates the trustpoint named ruckus.

```
device# configure terminal
device(config)# pki authenticate ruckus
```

# pki cert-validate

Determines if a trustpoint has been successfully authenticated.

## Syntax

**pki cert-validate** { *trustpoint* }

## Parameters

*trustpoint*

Designates the name of the trustpoint to be checked.

## Modes

Global configuration mode.

## Examples

The following example confirms that the trustpoint abcd has been successfully authenticated.

```
device# configure
device(config)# pki cert-validate abcd
PKI: Successfully validated the local certificate for trustpoint: abcd
```

# pki enroll

Requests certificates from the certificate authority (CA) for each key pair of the router.

## Syntax

**pki enroll** { *trustpoint* }

**no pki enroll** { *trustpoint* }

## Command Default

The router is not enrolled on the CA trustpoint.

## Parameters

**trustpoint**

Specifies the name of the trustopint where the router is to be enrolled.

## Modes

Global configuration mode.

## Usage Guidelines

The no form of the command removes the certificates from the router.

## Examples

The following example enrolls the router on the CA with the truspoint name ruckus

```
device# configure terminal
device(config)# pki enroll ruckus
```

# pki-entity

Configures a PKI entity and enters a configuration sub-mode where you can define PKI end-user parameters.

## Syntax

**pki-entity** { *entity* }

**no pki-entity** { *entity* }

## Parameters

*entity*

Names the entity for which parameters are to be configured.

## Modes

Global configuration mode.

## Usage Guidelines

The no form of the command removes the PKI entity and its configured parameters.

PKI entity configuration is used for auto-enrollment only.

## Examples

The following example enters PKI-entity configuration submode and configures PKI parameters.

```
device# configure terminal
device(config)# pki entity entity1
device(config-pki-entity-entity1)# common-name "tester1"
device(config-pki-entity-entity1)# country-name "IN"
device(config-pki-entity-entity1)# state-name "KA"
device(config-pki-entity-entity1)# org-unit-name "FI"
device(config-pki-entity-entity1)# org-name "BRCD"
device(config-pki-entity-entity1)# email-id "user@arris.com"
device(config-pki-entity-entity1)# location "BG"
device(config-pki-entity-entity1)#
device(config-pki-entity-entity1)# exit
```

# pki profile-enrollment

Enters PKI enrollment configuration submode, where you can onfigure PKI enrollment parameters.

## Syntax

**pki profile-enrollment** { *profile* }

**no pki profile-enrollment** { *profile* }

## Parameters

*profile*

Designates the name of the profile for which parameters are configured.

## Modes

Global configuration mode.

## Usage Guidelines

The no form of the command removes the profile and its configured parameters.

## Examples

The following example configures PKI enrollment profile profile1.

```
device# configure terminal
device(config)# pki profile-enrollment profile1
device(config-pki-profile-enrollment-profile1)# authentication-url http://WINN6C3R0LUDAJ.
englab.arris.com/CertSrv/mscep/mscep.dll
device(config-pki-profile-enrollment-profile1)# authentication-command WINN6C3R0LUDAJ.
englab.arris.com_englab-WIN-N6C3R0LUDAJ-CA-15
device(config-pki-profile-enrollment-profile1)# enrollment-url http://WINN6C3R0LUDAJ.
englab.arris.com/CertSrv/mscep/mscep.dll
device(config-pki-profile-enrollment-profile1)# password DB6E1F091AEF0244
device(config-pki-profile-enrollment-profile1)# exit
```

# pki trustpoint

Enters PKI trustpoint configuration mode, where PKI CA parameters can be configured.

## Syntax

**pki trustpoint** { *trustpoint* }

**no pki trustpoint** { *trustpoint* }

## Parameters

*trustpoint*

Names the trustpoint for which parameters are configured.

## Modes

Global configuration mode.

## Usage Guidelines

The no form of the command removes the trustpoint and its parameters.

## Examples

The following example configures the PKI trustpoint trust1.

```
device# configure terminal
device(config)# pki trustpoint trust1
device(config-pki-trustpoint-trust1)# auto-enroll
device(config-pki-trustpoint-trust1)# enrollment retry-period 2
device(config-pki-trustpoint-trust1)# enrollment profile profile1
device(config-pki-trustpoint-trust1)# pki-entity entity1
device(config-pki-trustpoint-trust1)# eckeypair key-label eckeyAuto
device(config-pki-trustpoint-trust1)# fingerprint 36:0c:92:6e:df:b2:72:eb:59:e8:63:73:2a:98:a8:91:cb:
50:94:d9
device(config-pki-trustpoint-trust1)# ocsp http post
device(config-pki-trustpoint-trust1)# exit
```

The following example shows the configuration options available in PKI trustpoint configuration sub-mode.

```
device(config-pki-trustpoint-trust2)# ?
auto-enroll                       To send enrollment message to the CA and local

                                 certificates.
clear                             Clear table/statistics/keys.
crl-query                         To set crl query url.
crl-update-time                   To set the CRL update period.
eckeypair                         To specify which ec keypair to use during
                                  enrollment.
end                               End Configuration level and go to Privileged
                                  level.
enrollment                        To set enrollment retry count, retry period or

                                 profile.
exit                              Exit current level.
extended-key-usage                To set or unset extended key usage parameters.
fingerprint                       To set fingerprint of the CA.
local-certificate                 URL of the local certificate.
no                                Undo/disable commands.
ocsp                              To set http method for ocsp request.
ocsp-url                          To set the oscp url.
pki-entity                        The PKI entity parameter to be used while enrolling

                                 to the CA.
quit                              Exit to User level.
revocation-check                  To specify which method to be followed for revocation
                                  check.
rsakeypair                        To specify which rsa keypair to use during
                                  enrollment.
show                              Show system information.
write                             Write running configuration to flash or terminal.
<cr>
device(config-pki-trustpoint-trust2)#
```

# poison-local-routes

Configures the device to avoid routing loops by advertising local RIP or RIPng routes with a cost of 16 (infinite or unreachable) when these routes go down.

## Syntax

**poison-local-routes**

**no poison-local-routes**

## Command Default

By default, RIP or RIPng routers add a cost of 1 to RIP or RIPng routes advertised to neighbors.

## Modes

RIP router configuration mode or RIPng router configuration mode.

## Usage Guidelines

Use the **no** form of the poison-local-routes command to disable these poison route updates for local routes that go down.

## Examples

The following example configures the RIP router to trigger an update to advertise local RIP routes as unreachable when they go down.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# poison-local-routes
```

The following example configures the RIPng router to trigger an update when local routes go down to advertise them as unreachable.

```
device# configure terminal
device(config)# ipv6 router rip
device(config-ripng-router)# poison-local-routes
```

# poison-reverse

Enables poison reverse loop prevention, either globally or on an individual interface, by assigning an "unreachable" cost to a route before advertising it on the interface where the route was learned. The global command can be used for RIP or RIPng routes.

## Syntax

**poison-reverse**

**ip rip poison-reverse**

**no poison-reverse**

**no ip rip poison-reverse**

## Command Default

By default, split horizon loop prevention is in effect. Split horizon does not advertise a route on the same interface as the one on which the device learned the route.

## Modes

RIP router configuration mode, RIPng router configuration mode, or interface configuration mode

## Usage Guidelines

The **no** form of the command disables poison reverse loop prevention.

Either poison reverse or split horizon loop prevention is always in effect on an interface enabled for RIP. When poison reverse is disabled, split horizon loop prevention is applied.

## Examples

The following command enables poison reverse loop prevention for RIP on a device.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# poison-reverse
```

The following example disables poison reverse and re-asserts split horizon loop prevention for RIP on the device.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# no poison-reverse
```

The following example enables poison reverse for RIP routes on Ethernet interface 1/2/3.

```
device# configure terminal
device(config)# interface ethernet 1/2/3
device(config-if-e10000-1/2/3)# ip rip poison-reverse
```

The following example enables poison reverse for RIPng on a device.

```
device# configure terminal
device(config)# ipv6 router rip
device(config-ripng-router)# poison-reverse
```

# port security

Enters port security configuration mode.

## Syntax

**port security**

## Modes

Global configuration mode

## Usage Guidelines

Use the **enable** command to enable port security.

## Examples

The following example shows how to enter port security configuration mode.

```
device(config)# port security
device(config-port-security)#
```

# port-down-authenticated-mac-cleanup

Enables forced reauthentication of the hosts if all the ports on the device go down.

## Syntax

**port-down-authenticated-mac-cleanup**

**no port-down-authenticated-mac-cleanup**

## Command Default

Forced reauthentication of hosts is enabled.

## Modes

Web Authentication configuration mode

## Usage Guidelines

When the command is enabled, the device checks the link state of all ports that are members of the Web Authentication VLAN. If the state of all the ports is down, then the device forces all authenticated hosts to reauthenticate. However, hosts that were authenticated using the **add mac** command will remain authenticated; they are not affected by the **port-down-authenticated-mac-cleanup** command.

The **no** form of the command removes forced reauthentication of the hosts.

## Examples

The following example enables forced reauthentication of all hosts when all the ports are down.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# port-down-authenticated-mac-cleanup
```

# port-name

Configures the names of individual ports or a group of ports.

## Syntax

**port-name** *text*

**no port-name** *text*

## Command Default

A port name is not configured.

## Parameters

*text*

Configures the name of the port or the name of a range of ports. The name is an alphanumeric string and can be up to 255 characters long.

## Modes

Interface configuration mode

## Usage Guidelines

You can assign a port name to physical ports, virtual interfaces, and loopback interfaces. The port name can contain blank spaces. The port name can also contain special characters, but the percentage character (%) is dropped if it is the last character in the port name.

The **no** form of the command removes the assigned port name.

## Examples

The following example assigns a name to a port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port-name Marsha
```

The following example assigns a name to a range of ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/10
device(config-mif-1/1/1-1/1/10)# port-name connected-to-the nearest device
```

The following example assigns a name to multiple ports.

```
device(config)# interface ethernet 1/1/1 ethernet 1/1/5 ethernet 1/1/7
device(config-mif-1/1/1,1/1/5,1/1/7)# port-name connected-to-the nearest device
```

# port-name (LAG)

Assigns a port name to an individual port in a LAG.

## Syntax

**port-name** *name* **ethernet** *stackid/slot/port*

**no port-name** *name* **ethernet** *stackid/slot/port*

## Command Default

A port name is not assigned to an individual port within a LAG.

## Parameters

*name*
Specifies the name of an individual port in a LAG. The name can be up to 255 characters in length.

**ethernet** *stackid/slot/port*
Specifies the Ethernet port to which the name must be assigned.

## Modes

LAG configuration mode

## Usage Guidelines

When creating a port name in a LAG, you can use all uppercase or lowercase characters, as well as digits. Special characters (such as $, %, ', -, ., @, ~, `, !, (, ), {, }, ^, #, and &) are valid. You can use spaces in the port name as long as you enclose the name in double quotation marks. For example, to specify a port name that contains spaces, enter a string similar to the following example: "a long and lengthy port name".

**NOTE**
A port name with spaces must be enclosed within double quotation marks.

The **no** form of the command removes the name assigned to the individual port.

## Examples

The following example shows how to assign a name to a port in a LAG.

```
device(config)# lag "test" dynamic id 1
device(config-lag-test)# ports ethernet 1/1/1 to 1/1/3
device(config-lag-test)# port-name "lag1" ethernet 1/1/1
```

# port-statistics-reset-timestamp enable

Enables the display of the elapsed timestamp information in the output of the **show statistics** command.

## Syntax

**port-statistics-reset-timestamp enable**

**no port-statistics-reset-timestamp enable**

## Command Default

The elapsed time after the recent reset of the port statistics counters is not displayed in the **show statistics** command output.

## Modes

Global configuration mode

## Usage Guidelines

The elapsed time is calculated as the time between the most recent reset of the port statistics counters and the time when the **show statistics** command is executed.

The **port-statistics-reset-timestamp enable** command enables the display of the elapsed timestamp information for all the ports in the output of the **show statistics** command.

The **no** form of the command removes the display of the elapsed time after the most recent reset of the port statistics counters in the **show statistics** command output.

## Examples

The following example enables the display of the elapsed time between the most recent reset of the port statistics counters and the time when the **show statistics** command is executed.

```
device (config)# port-statistics-reset-timestamp enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# ports

Adds ports in a LAG.

## Syntax

**ports ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ... ]

**no ports ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] ... ]

## Command Default

No ports are added to the LAG.

## Parameters

**ethernet** *stackid/slot/port*

Adds an Ethernet interface to a LAG.

**to** *stackid/slot/port*

Adds a range of Ethernet interfaces to the LAG.

## Modes

LAG configuration mode

## Usage Guidelines

A static or dynamic LAG can have 1 to 8 or 1 to 16 ports (depending on the device you are using) of the same type and speed that are on any interface module within the Ruckus chassis. A keep-alive LAG consists of only one port.

Ports can be added to an undeployed LAG or to a currently deployed LAG. If removal of a port will result in the trunk threshold value becoming greater than the number of ports in the LAG, the port deletion will be rejected. When you remove a port from a deployed LAG, the port is disabled automatically.

The **no** form of the command removes the ports from a LAG.

## Examples

The following example shows how to configure a static LAG with two ports.

```
device(config)# lag blue static id 1
device(config-lag-blue)# ports ethernet 1/3/1 ethernet 1/3/2
```

The following example adds a range of ports to the LAG.

```
device(config)# lag blue static
device(config-lag-blue)# ports ethernet 1/3/1 to 1/3/4
```

The following example adds a range of ports from one interface module and an individual port from another interface module to the LAG.

```
device(config)# lag blue static
device(config-lag-blue)# ports ethernet 1/3/1 to 1/3/4 ethernet 1/2/2
```

# pre-shared-key

Configures the pre-shared MACsec key on the interface.

## Syntax

**pre-shared-key** *key-id* **key-name** *hex-string*

**no pre-shared-key** *key-id* **key-name** *hex-string*

## Command Default

No pre-shared MACsec key is configured on the interface.

## Parameters

*key-id*

Specifies the 32 hexadecimal value used as the Connectivity Association Key (CAK).

**key-name** *hex-string*

Specifies the name for the CAK key. Use from 2 through 64 hexadecimal characters to define the key name.

## Modes

dot1x-mka interface mode

## Usage Guidelines

The **no** form of the command removes the pre-shared key from the interface.

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

The pre-shared key is required for communications between MACsec peers.

## Examples

The following example configures MKA group test1 and assigns the MACsec pre-shared key with a name beginning with 96437a93 and with the value shown, to port 2, slot 3 on the first device in the stack.

```
device(config)#dot1x-mka-enable
device(config-dot1x-mka)# mka-cfg-group test1
device(config-dot1x-mka-group-test1)# key-server-priority 5
device(config-dot1x-mka-group-test1)# macsec cipher-suite gcm-aes-128
device(config-dot1x-mka-group-test1)# macsec confidentiality-offset 30
device(config-dot1x-mka-group-test1)# exit
device(config-dot1x-mka)# enable-mka ethernet 1/3/2
device(config-dot1x-mka-1/3/2)# mka-group test1
device(config-dot1x-mka-1/3/2)# pre-shared-key 135bd758b0ee5c11c55ff6ab19fdb199 key-name
96437a93ccf10d9dfe347846cce52c7d
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Adds MACsec support on ICX 7650 devices. |

# prefix-list

Associates an IPv6 prefix list with a Router Advertisement (RA) guard policy.

## Syntax

**prefix-list** *name*

**no prefix-list** *name*

## Parameters

*name*

Specifies the name of the IPv6 prefix list to associate with the RA guard policy.

## Modes

RA guard policy configuration mode

## Usage Guidelines

This command associates an IPv6 prefix list with an RA guard policy so that only the RAs that have the given prefix are forwarded. You must provide the name of an IPv6 prefix list already configured using the **ipv6 prefix-list** command. For more information on configuring an IPv6 prefix list using the **ipv6 prefix-list** command, see the *FastIron Ethernet Switch Layer 3 Routing Configuration Guide* .

Only one prefix list can be associated with an RA guard policy. If the command is configured twice with different prefix lists, the latest configured prefix list is associated with the RA guard policy.

**no**

## Examples

The following example associates an IPv6 prefix list with an RA guard policy:

```
device(config)# ipv6 prefix-list raguard-prefix1
device(config)# ipv6 raguard policy p1
device(config-ipv6-RAG-policy p1)# prefix-list raguard-prefix1
```

## Related Commands

neighbor prefix-listneighbor prefix-listneighbor prefix-list

# prefix-list (RIP)

Applies a pre-configured prefix list to permit or deny RIP routes globally.

## Syntax

**prefix-list** *name* { **in** | **out** }

**no prefix-list** *name* { **in** | **out** }

**ip rip prefix-list** *name* { **in** | **out** }

**no ip rip prefix-list** *name* { **in** | **out** }

## Parameters

*name*
: Specifies the pre-configured prefix list to be applied.

**in**
: Applies the specified prefix list to routes the device learns from its neighbors.

**out**
: Applies the specified prefix list to routes the device advertises to its neighbors.

## Modes

RIP router configuration mode.

## Usage Guidelines

The **no** form of the command removes the prefix filter.

Prefix lists must be configured with the **ip prefix-list** command before they are applied.

The **ip rip prefix-list** command can be used to apply a prefix list at the interface level.

## Examples

The following command globally applies the prefix list named list1 to routes that the RIP router learns from its neighbors.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# prefix-list list1 in
```

The following command applies the prefix list named test1 to RIP routes advertised on Ethernet interface 1/1/2.

```
device# configure terminal
device(config)# interface ethernet 1/1/2
device(config-if-e10000-1/1/2)# ip rip prefix-list test1 out
```

# preforwarding-time

Configures the preforwarding time interval, the time a port will remain in the preforwarding state before changing to the forwarding state.

## Syntax

**preforwarding-time** *milliseconds*

**no preforwarding-time** *milliseconds*

## Command Default

The default preforwarding time interval is 300 milliseconds.

## Parameters

*milliseconds*

The preforwarding time interval in milliseconds. The range is from 200 through 30000 milliseconds.

## Modes

MRP configuration mode

## Usage Guidelines

The preforwarding time interval must be at least twice the value of the hello time or a multiple of the hello time.

When MRP is enabled, all ports begin in the preforwarding state.

An interface changes from the preforwarding state to the forwarding state when the port preforwarding time expires. This occurs if the port does not receive a Ring Health Packet (RHP) from the master, or if the forwarding bit in the RHPs received by the port is off (indicating a break in the ring). The port heals the ring by changing its state to forwarding. If a member port in the preforwarding state does not receive an RHP within the preforwarding time, the port assumes that a topology change has occurred and changes to the forwarding state.

The secondary port on the master node changes to the blocking state if it receives an RHP, but changes to the forwarding state if the port does not receive an RHP before the preforwarding time expires. A member node preforwarding interface also changes from preforwarding to forwarding if it receives an RHP whose forwarding bit is on.

If Unidirectional Link Detection (UDLD) is also enabled on the device, Ruckus recommends that you set the MRP preforwarding time slightly higher than the default of 300 ms; for example, to 400 or 500 ms.

The **no** form of the command sets the preforwarding time interval to the default.

# Examples

The following example shows how to configure the preforwarding time to 400 milliseconds.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-1)# preforwarding-time 400
```

# prf

Configures a pseudorandom function (PRF) for an Internet Key Exchange version 2 (IKEv2) proposal.

## Syntax

**prf** { **sha256** | **sha384** }

**no prf** { **sha256** | **sha384** }

## Command Default

The default algorithm is SHA-384.

## Parameters

**sha256**

Specifies SHA-2 family 256-bit (HMAC variant) as the hash algorithm.

**sha384**

Specifies SHA-2 family 384-bit (HMAC variant) as the hash algorithm.

## Modes

IKEv2 proposal configuration mode

## Usage Guidelines

This hash algorithm is used to generate key material during IKEv2 SA negotiations.

Both algorithms may be configured for an IKEv2 proposal.

When only one PRF algorithm is configured for an IKEv2 proposal, removing it restores the default configuration.

The **no** form of the command removes the specified PRF algorithm configuration.

## Examples

The following example shows how to configure SHA-256 as the hash algorithm for an IKEv2 proposal named ikev2_prop.

```
device(config)# ikev2 proposal ikev2_prop
device(config-ikev2-proposal-ikev2_prop)# prf sha256
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# priority

Configures a priority value for the device. This value is used along with other factors to determine controller election if a stack failover or merge occurs.

## Syntax

**priority** *num*

**no priority**

## Command Default

The priority value for the active controller and standby device is 128.

## Parameters

*num*

Possible values are 0 to 255. Lower values assign a lower priority to the device, and higher values assign a higher priority to the device.

## Modes

Stack unit configuration mode

## Usage Guidelines

The **no** form of the command restores the default priority value to the device (128). You do not have to specify the default value when using the **no** form.

A unit that has a relatively high priority value is more likely to be elected to be the active controller.

When you change the priority value assigned to a stack unit, the value takes effect immediately but does not affect the active controller until the next reset.

When the active and standby controller have the same priority value, other factors affect controller election, such as up-time and number of members controlled.

## Examples

The following example assigns a priority value of 130 to stack unit 1.

```
device(Config)# stack unit 1
device(Config-unit-1)# priority 130
```

## History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced. |

# priority-flow-control

Enables priority flow control (PFC) on a priority group.

## Syntax

**priority-flow-control** *priority-group-number*

**no priority-flow-control** *priority-group-number*

## Command Default

PFC is disabled globally.

## Parameters

*priority-group-number*
Specifies a priority group. The range is from 0 through 3.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on Ruckus ICX 7250, Ruckus ICX 7450, and Ruckus ICX 7750 devices. This command is not supported on Ruckus ICX 7150 or Ruckus ICX 7650 devices.

To enable global PFC, symmetrical flow control (SFC) must be disabled.

You must enable PFC globally before you configure it for priority groups. Enabling PFC on a priority group enables PFC on all the ports.

PFC and 802.3x flow control are mutually exclusive. Configuring the **priority-flow-control** command disables 802.3x in both transmit and receive directions.

PFC is not supported for ports across stack units on Ruckus ICX 7750 devices.

The **no** form of this command restores the default flow-control settings.

## Examples

The following example enables PFC for priority group 2:

```
Device(config)# priority-flow-control enable
Device(config)# priority-flow-control 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |
| 08.0.20 | This command was modified. Specifying a priority group no longer enables PFC on all ports. |
| 08.0.60 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7150. |
| 08.0.70 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7650. |

# priority-flow-control enable

Enables priority flow control (PFC) globally or on an individual port.

## Syntax

**priority-flow-control enable**

**no priority-flow-control enable**

## Command Default

PFC is disabled (globally and on all ports).

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

This command is supported only on Ruckus ICX 7250, Ruckus ICX 7450, and Ruckus ICX 7750 devices. This command is not supported on Ruckus ICX 7150 or Ruckus ICX 7650 devices.

To enable global PFC, symmetrical-flow-control (SFC) must be disabled.

You must enable PFC globally before you configure it for priority groups.

In global configuration mode, configuring the **priority-flow-control enable** command enables PFC globally; in interface configuration mode, configuring the command enables PFC on a port. You can configure the **priority-flow-control enable** command in interface configuration mode to enable both PFC transmit and receive, which means PFC is both honored and generated. PFC must be enabled on at least one priority group before you can configure the **priority-flow-control enable** command on an interface.

Priority flow control and 802.3x flow control are mutually exclusive; therefore, configuring the **priority-flow-control enable** command disables 802.3x in both transmit and receive directions.

The **no** form of the command restores the default flow-control settings in global configuration mode and disables PFC on the interface in interface configuration mode.

## Examples

The following example enables PFC globally.

```
Device(config)# priority-flow-control enable
```

The following example enables PFC on an interface.

```
Device(config-if-e10000-1/1/1)# priority-flow-control enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |
| 08.0.20 | This command was modified to add enabling PFC on a port. |
| 08.0.60 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7150. |
| 08.0.70 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7650. |

# privilege

Configures the management privilege access level of a command.

## Syntax

**privilege** *command-mode* **level** *privilege-level command-string*

**no privilege** *command-mode* **level** *privilege-level command-string*

## Parameters

*command-mode*

Specifies the command mode (CLI level) of the command for which the access level is to be enhanced. The following values are available:

- **exec** - EXEC level; for example, device> or device#
- **configure** - global configuration level; for example, device(config)#
- **interface** - Interface level; for example, device(config-if-e1000-1/2/3)#
- **loopback-interface** - Loopback interface configuration sub-mode
- **virtual-interface** - Virtual-interface configuration sub-mode; for example, device(config-vif-6)#
- **dot1x** - 802.1X configuration sub-mode
- **ipv6-access-list** - IPv6 access list configuration sub-mode
- **rip-router** - RIP router configuration sub-mode; for example, device(config-rip-router)#
- **ospf-router** - OSPF router configuration sub-mode; for example, device(config-ospf-router)#
- **dvmrp-router** - DVMRP router configuration sub-mode; for example, device(config-dvmrp-router)#
- **pim-router** - PIM router configuration sub-mode; for example, device(config-pim-router)#
- **bgp-router** - BGP4 router configuration sub-mode; for example, device(config-bgp-router)#
- **vrrp-router** - VRRP configuration sub-mode
- **gvrp** - GVRP configuration sub-mode
- **trunk** - trunk configuration sub-mode
- **port-vlan** - Port-based VLAN configuration sub-mode; for example, device(config-vlan)#
- **protocol-vlan** - Protocol-based VLAN configuration sub-mode

Enter **?** to check for available interface subtypes.

**level** *privilege-level*

Specifies the number of the management privilege level you are augmenting. Valid values are as follows:

- 0 - Super User level (full read-write access)
- 4 - Port Configuration level
- 5 - Read Only level.

*command-string*

Specifies the command you want to assign the specified privilege level.
Enter **?** at the command prompt of a CLI level to display the list of commands at that level.

# Modes

Global configuration mode

# Usage Guidelines

Each management privilege level provides access to specific areas of the CLI by default. You can grant additional access to a privilege level on an individual command basis. To grant the additional access, specify the privilege level you are enhancing, the CLI level that contains the command, and the individual command.

Super User management privilege provides access to all commands and displays.

Port Configuration management privilege provides access to the following levels:

- The User EXEC
- Privileged EXEC
- The port-specific parts of global configuration
- All interface configuration.

Read Only management privilege level gives access to the following levels:

- User EXEC
- Privileged EXEC

**NOTE**
The **privilege** command applies only to management privileges for the CLI.

The **no** form of the **privilege** command removes the configuration and resets default privilege levels.

# Examples

The following example shows how to enhance the Port Configuration privilege level so users also can enter IP commands at the global configuration level.

All users with Port Configuration privileges receive the enhanced access after the command is entered. Executing this command will enable users who log in with valid Port Configuration level user names and passwords to execute commands that start with "ip" at the global configuration level.

```
device# configure terminal
device(config)# privilege configure 4 ip
```

# profile-config

Configures the port buffer, queue buffer, port descriptor, and queue descriptor for a port.

## Syntax

**profile-config** { **port-buffers** *buffer-number* | **port-descriptors** *descriptor-number* | **port-type** { **0** | **1** | **2** | **3** } | **queue-buffers** *egress-queue-number buffer-number* | **queue-descriptors** *egress-queue-number descriptor-number* }

**no profile-config** { **port-buffers** *buffer-number* | **port-descriptors** *descriptor-number* | **port-type** { **0** | **1** | **2** | **3** } | **queue-buffers** *egress-queue-number buffer-number* | **queue-descriptors** *egress-queue-number descriptor-number* }

## Command Default

The default port type is set to 1 Gbps.

The default buffers and descriptors are set according to the port type.

## Parameters

**port-buffers** *buffer-number*
Configures the maximum buffer limit for the port.

**port-descriptors** *descriptor-number*
Configures the maximum descriptor limit for the port.

**port-type**
The port type for the user-configurable buffer profile.

**0**
Specifies the port type as 1 Gbps, 10 Gbps, or 40 Gbps.

**1**
Specifies the port type as 1 Gbps.

**2**
Specifies the port type as 10 Gpbs.

**3**
Specifies the port type as 40 Gbps.

**queue-buffers**
Configures the maximum buffer limit for the queues.

*egress-queue-number*
Specifies the egress queue number (0 through 7).

*buffer-number*
Specifies the buffer number.

**queue-descriptors**
Configures the maximum descriptor limit for the queues.

*descriptor-number*
Specifies the descriptor number.

# Modes

Buffer profile configuration mode

# Usage Guidelines

To configure a user-configurable profile for 10 Gbps ports, the 10 Gbps port type must be explicitly provided by the **port-type** option. Modifications to buffers and descriptors of a port and its queues take effect dynamically.

When the profile type is configured as all 1 Gbps, 10 Gbps, and 40 Gbps ports, the default buffers and descriptors will be set according to the port type; that is, all 1 Gbps ports use 1 Gbps defaults and 10 Gbps ports use 10 Gbps defaults. If you configure a port and its queue with egress buffer and descriptor limits, then the configured limits are used for both 1 Gbps and 10 Gbps ports.

Port type modification resets the profile to its default value. All the port and queue buffers and descriptors will be set to either 1 Gbps or 10 Gbps defaults as per the configuration, which means all the user configurations for the port and its queues will be lost.

**NOTE**
Port type modifications on an active profile are not allowed.

The **no** form of the command with the **port-type** option sets the profile port type to 1 Gbps.

# Examples

The following example sets the port type to 10 Gbps.

```
device(config)# qd-buffer-profile 1
device(qd-profile-1)# profile-config port-type 3
```

The following example configures the port buffers.

```
device(config)# qd-buffer-profile 1
device(qd-profile-profile1)# profile-config port-buffers 8000
```

The following example configures the port descriptors.

```
device(config)# qd-buffer-profile 1
device(qd-profile-profile1)# profile-config port-descriptors 8000
```

The following example configures the queue buffer.

```
device(config)# qd-buffer-profile 1
device(qd-profile-profile1)# profile-config queue-buffers 2 600
```

The following example configures the queue descriptors.

```
device(config)# qd-buffer-profile 1
device(qd-profile-profile1)# profile-config queue-descriptors 2 600
```

# proposal (IKEv2)

Configures an Internet Key Exchange version 2 (IKEv2) proposal for an IKEv2 policy.

## Syntax

**proposal** *name*

**no proposal** *name*

## Command Default

The default IKEv2 proposal (**def-ike-prop**) is configured for an IKEv2 policy.

## Parameters

*name*

Specifies the name of an IKEv2 proposal.

## Modes

IKEv2 policy configuration mode

## Usage Guidelines

At least one IKEv2 proposal must be configured for an IKEv2 policy.

Multiple IKEv2 proposals may be configured for an IKEv2 policy.

When only one IKEv2 proposal is configured for an IKEv2 policy, removing it restores the default configuration.

The **no** form of the command removes the specified IKEv2 proposal from the IKEv2 policy configuration.

## Examples

The following example shows how to configure an IKEv2 proposal named ikev2_proposal1 for an IKEv2 policy named ikev2_policy1.

```
device# configure terminal
device(config)# ikev2 policy ikev2_policy1
device(config-ike-policy-ikev2_policy1)# proposal ikev2_proposal1
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# proposal (ipsec)

Configures an IP security (IPsec) proposal for an IPsec profile.

## Syntax

**proposal** *name*

**no proposal** *name*

## Command Default

The default IPsec proposal (**def-ipsec-prop**) is configured for an IPsec profile.

## Parameters

*name*

Specifies the name of an IPsec proposal.

## Modes

IPsec profile configuration mode

## Usage Guidelines

IPsec is supported only on ICX 7450 devices.

Multiple IPsec proposals may be configured for an IPsec profile.

When only one IPsec proposal is configured for an IPsec profile, removing it restores the default configuration.

The **no** form of the command removes the specified IPsec proposal from the IPsec profile configuration.

## Examples

The following example shows how to configure an IPsec proposal named ipsec_proposal1 for an IPsec profile named ipsec_profile1.

```
device# configure terminal
device(config)# ipsec profile ipsec_profile1
device(config-ipsec-profile-ipsec_profile1)# proposal ipsec_proposal1
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# protected

Configures VRF traffic protection for an Internet Key Exchange version 2 (IKEv2) profile.

## Syntax

**protected** *vrf*

**no protected** *vrf*

## Parameters

*vrf*

Specifies the name of the VRF to be protected.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

When the tunnel VRF and the protected VRF do not match, an IKEv2 session is not initiated.

The **no** form of the command removes the specified VRF traffic protection configuration for the IKEv2 profile.

## Examples

The following example shows how to configure an IKEv2 profile named test to protect traffic for a VRF named red.

```
device(config)# ikev2 profile test
device(config-ikev2-profile-test)# protected red
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# protected-port

Configures a port as protected, restricting communication among such ports at the system level, providing isolation to end hosts.

## Syntax

**protected-port**

**no protected-port**

## Command Default

Protected port is not enabled.

## Modes

Interface configuration mode

## Usage Guidelines

Use the **no** form of this command to disable the protected port feature.

The following configurations are supported with the protected port feature:

- Port MAC security
- 802.1x security
- DHCP snooping
- Control protocols
- Aggregated ports (LAGs)

The following should not be configured as protected ports:

- Uplink ports
- DHCP server ports
- ARP inspection trusted ports
- DHCP snooping trusted ports
- Ports on an active xSTP path in a device
- IGMP/MLD snooping router ports
- IGMP/MLD source ports

In addition, it is recommended that multiple ports (MIF) mode be configured.

The following features are not supported on protected ports:

- Layer 3 interfaces (IP addresses are not supported)
- Mirror or monitor ports
- Private VLAN (PVLAN)
- PVLAN extension to protected-port switches

- Virtual Ethernet (VE) and group VE interfaces
- Loopback interfaces
- Management interfaces
- OpenFlow ports
- SPX provider edge (PE) ports
- SPX ZTP–enabled ports
- Multi-Chassis Trunk (MCT)

# Examples

The following example enables protected port on a single interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# protected-port
```

The following example enables protected port on multiple ports in MIF mode.

```
device# configure terminal
device(config)# interface ethernet 2/1/1 ethernet 3/1/1
device(config-if-e1000-2/1/1,3/1/1)# protected-port
```

The following example disables protected port for the previous example.

```
device# configure terminal
device(config)# interface ethernet 2/1/1 ethernet 3/1/1
device(config-if-e1000-2/1/1,3/1/1)# no protected-port
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# prune-timer

Configures the time a PIM device maintains a prune state for a forwarding entry.

## Syntax

**prune-timer** *seconds*

**no prune-timer** *seconds*

## Command Default

The prune time is 180 seconds.

## Parameters

*seconds*

Specifies the interval in seconds. The range is 60 through 3600 seconds. The default is 180 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the default prune time, 180 seconds.

The first received multicast interface is forwarded to all other PIM interfaces on the device. If there is no presence of groups on that interface, the leaf node sends a prune message upstream and stores a prune state. This prune state travels up the tree and installs a prune state. A prune state is maintained until the prune timer expires or a graft message is received for the forwarding entry.

## Examples

This example configures a PIM prune timer to 90 seconds.

```
Device(config)# router pim
Device(config-pim-router)# prune-timer 90
```

# prune-wait

Configures the time a PIM device waits before stopping traffic to neighbor devices that do not want the traffic.

## Syntax

**prune-wait** *seconds*

**no prune-wait**

## Command Default

The prune wait time is 3 seconds.

## Parameters

*seconds*

Specifies the wait time in seconds. The range is 0 through 30 seconds. The default is 3 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

A smaller prune wait value reduces flooding of unwanted traffic. A prune wait value of 0 causes the PIM device to stop traffic immediately upon receiving a prune message.

If there are two or more neighbors on the physical port, you should not configure the **prune-wait** command because one neighbor may send a prune message while the other sends a join message at the same time, or within less than 3 seconds.

The **no** form of this command restores the default prune wait time of 3 seconds.

## Examples

This example configures the prune wait time to 0 seconds.

```
device(config)# router pim
device(config-pim-router)# prune-wait 0
```

# pvlan mapping

Creates a Private VLAN domain with Primary-Secondary pair. Maps promiscuous and host ports to VLAN based forwarding in PVLAN domain.

## Syntax

**pvlan mapping** *vlan-id* [ **ethernet** *stackid/slot/port* | **lag** *decimal* ]

**no pvlan mapping** *vlan-id* [ **ethernet** *stackid/slot/port* | **lag** *decimal* ]

## Command Default

PVLAN mapping is not configured.

## Parameters

*vlan-id*

Secondary VLAN ID to be mapped to Primary.

**ethernet** *stackid**/**slot**/**port*

Specifies the ethernet interface ( stack ID or Slot or port) to which the secondary VLAN is to be mapped.

**lag** *decimal*

Specifies the LAG interface (decimal) to which the secondary VLAN is to be mapped.

## Modes

VLAN configuration mode

Privileged EXEC

## Usage Guidelines

This command will map the secondary VLAN to promiscuous port or LAG. Issue this command only on primary VLAN.Secondary VLAN to be mapped should be VALID.Port or LAG with which secondary VLAN is to be mapped should be a member of Primary VLAN.

The **no** form of the command removes the mapping of Secondary VLAN with promiscuous port or LAG.

# Examples

The following example shows how to configure PVLAN mapping on an Ethernet interface.

```
device(config)#lag ISL dynamic id 1
device(config-lag-ISL)# ports ethe 1/2/3 ethe 1/2/4
LAG ISL deployed successfully!
device(config-lag-ISL)#!
device(config-lag-ISL)#lag promiscuous dynamic id 2
device(config-lag-promiscuous)# ports ethe 1/2/1 ethe 1/2/2
LAG promiscuous deployed successfully!
device(config-lag-promiscuous)#!
device(config-lag-promiscuous)#lag wallplate-1 dynamic id 3
device(config-lag-wallplate-1)# ports ethe 1/1/1 to 1/1/4
LAG wallplate-1 deployed successfully!
devoce(config-lag-wallplate-1)#!
device(config-lag-wallplate-1)#lag wallplate-2 dynamic id 4
device(config-lag-wallplate-2)# ports ethe 1/1/5 to 1/1/8
LAG wallplate-2 deployed successfully!
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to support LAG ID options. |
| 08.0.70 | This command was modified to support mapping of secondary VLAN to promiscuous port or LAG. |

# pvlan pvlan-trunk

Creates a Private VLAN domain with Primary-Secondary pair. Maps ISL and host ports to VLAN based forwarding in PVLAN domain.

## Syntax

**pvlan pvlan-trunk** *num* **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ]... ]

**no pvlan pvlan-trunk** *num* **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ]... ]

**pvlan pvlan-trunk** *num* **lag** *decimal* [ **to** *decimal* | [ **lag** *decimal* **to** *decimal* | **lag** *decimal* ]... ]

**no pvlan pvlan-trunk** *num* **lag** *decimal* [ **to** *decimal* | [ **lag** *decimal* **to** *decimal* | **lag** *decimal* ]... ]

## Command Default

The inter-switch link for the primary VLAN is not configured.

## Parameters

*num*
Secondary VLAN ID to be mapped to Primary.

**ethernet** *unit/slot/port*
Specifies the ethernet interface ( stack ID or Slot or port) to which the secondary VLAN is to be mapped.

**to** *unit/slot/port*
Configures a range of Ethernet interfaces as the ISLs.

**lag** *decimal*
Specifies the LAG interface (decimal) to which the secondary VLAN is to be mapped.

**to** *decimal*
Configures a set of LAG virtual interfaces as the ISLs.

## Modes

VLAN configuration mode

Privileged EXEC mode

## Usage Guidelines

Issue this command only on Primary VLAN. Secondary VLAN for which an inter-switch link is to be created should be VALID. Port or LAG on which an inter-switch link is to be created should be a member of Primary VLAN.

The **no** form of the command removes the ISL port or LAG.

# Examples

The following example shows on Ethernet interfaces how to identify the ISL in the PVLAN.

```
device(config)# vlan 100
device(config-vlan-100)# tagged ethernet 1/1/10 to 1/1/11
device(config-vlan-100)# untagged ethernet 1/1/4
device(config-vlan-100)# pvlan type primary
device(config-vlan-100)# pvlan mapping 101 ethernet 1/1/4
device(config-vlan-100)# pvlan mapping 102 ethernet 1/1/4
device(config-vlan-100)# pvlan pvlan-trunk 101 ethernet 1/1/10 to 1/1/11
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to support the LAG ID option. |
| 08.0.61 | This command was modified to support mapping of secondary VLAN to promiscuous port or LAG. |

# pvlan type

Configures the PVLAN as a primary, isolated, or community PVLAN.

## Syntax

**pvlan type** { **community** | **isolated** | **primary** }

**no pvlan type** { **community** | **isolated** | **primary** }

## Command Default

The PVLAN type is not configured.

## Parameters

**community**
Creates a community PVLAN.

**isolated**
Creates an isolated PVLAN.

**primary**
Creates a primary PVLAN.

## Modes

VLAN configuration mode

## Usage Guidelines

The command configures the following PVLAN types:

- Community - Broadcasts and unknown unicasts received on community ports are sent to the primary port and also are flooded to the other ports in the community VLAN.
- Isolated - Broadcasts and unknown unicasts received on isolated ports are sent only to the primary port. They are not flooded to other ports in the isolated VLAN
- Primary - The primary PVLAN ports are "promiscuous". They can communicate with all the isolated PVLAN ports and community PVLAN ports in the isolated and community VLANs that are mapped to the promiscuous port.

For the primary VLAN, map the other PVLANs to the ports in the primary VLAN. VLAN identifiers configured as part of a PVLAN (primary, isolated, or community) should be consistent across the switched network. The same VLAN identifiers cannot be configured as a normal VLAN or a part of any other PVLAN.

LAG ports are not allowed as member ports of an isolated VLAN or community VLAN.

The **no** form of the command disables the PVLAN type.

# Examples

The following example shows how to configure the community PVLAN.

```
device(config)# vlan 901
device(config-vlan-901)# untagged ethernet 1/3/5 to 1/3/6
device(config-vlan-901)# pvlan type community
```

The following example shows how to configure a primary PVLAN.

```
device(config)# vlan 7
device(config-vlan-7)# untagged ethernet 1/3/2
device(config-vlan-7)# pvlan type primary
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was modified. |

# pvst-mode

Enables Per-VLAN Spanning Tree Plus (PVST+) support on a port immediately.

## Syntax

**pvst-mode**

**no pvst-mode**

## Command Default

PVST+ support is automatically enabled when the port receives a PVST BPDU.

## Modes

Interface configuration mode

## Usage Guidelines

This command cannot be executed concurrently with the **pvstplus-protect** command.

If you disable PVST+ support, the software still automatically enables PVST+ support if the port receives a BPDU with the PVST+ format.

The **no** form of the command disables the PVST+ support.

## Examples

The following example shows how to enable the PVST+ mode.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# pvst-mode
```

## History

| Release version | Command history |
|---|---|
| 08.0.30mb | The **pvstplus-protect** command restriction was added.. |

# pvstplus-protect

Prevents flooding and resulting port blocking on an interface when a Per-VLAN Spanning Tree Plus (PVST+) packet is received on a port configured for Multiple Spanning Tree Protocol (MSTP), blocking the PVST+ Bridge Protocol Data Unit (BPDU) and marking the port as ERR-DISABLED.

## Syntax

**pvstplus-protect**

**no pvstplus-protect**

## Command Default

PVST+ Protect is disabled.

## Modes

Interface configuration mode

## Usage Guidelines

This command cannot be executed concurrently with the **pvst-mode** command.

When you use the **pvstplus-protect** command, you must also use the global **errdisable recovery pvstplus-protect** command to enable ports to recovery from the error-disabled state.

The **no** form of the command disables PVST+ Protect.

## Examples

The following example enables PVST+ Protect on a single port.

```
device(config)# interface ethernet 1/1/1
device(config-if-1/1/1)# pvstplus-protect
```

The following example enables PVST+ Protect on a range of ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/4
device(config-mif-1/1/1-1/1/4)# pvstplus-protect
```

## History

| Release version | Command history |
|---|---|
| 8.0.30mb | This command was introduced. |

# qd-buffer

Configures the port buffers.

## Syntax

**qd-buffer** *device-num buffer-profile queue-depth* [ *priority-queue* ]

**no qd-buffer** *device-num buffer-profile quue-depth* [ *priority-queue* ]

## Command Default

Port buffers are not configured.

## Parameters

*device-num*
Specifies the device in the stacking unit. The device number starts from 1.

*buffer-profile*
Specifies the buffer profile: 1 for 1-Gbps ports, 2 for 10-Gbps ports, and 3 for VoIP ports.

*queue-depth*
Specifies the number of buffers to allocate.

*priority-queue*
Specifies the queue of the port. The range is from 0 through 7.

## Modes

Global configuration mode

## Usage Guidelines

The minimum limit for port buffers is 16. The maximum limit for the port buffer depends on the hardware device.

The **no** form of the command deletes the port buffers.

## Examples

The following example configures the port buffers.

```
device(config)# qd-buffer 1 2 76
```

The following example configures the queue buffers.

```
device(config)# qd-buffer 1 2 76 2
```

# qd-descriptor

Configures the allowable port descriptors.

## Syntax

**qd-descriptor** *device-num buffer-profile num-of-descriptors* [ *priority-queue* ]

**no qd-descriptor** *device-num buffer-profile num-of-descriptors* [ *priority-queue* ]

## Command Default

Port descriptors are not configured.

## Parameters

*device-num*
Specifies the device in the stacking unit. The device number starts from 0.

*buffer-profile*
Specifies the buffer profile. 1 for 1-Gbps ports and 2 for 10-Gbps ports.

*num-of-descriptors*
Specifies the number of descriptors to allocate.

*priority-queue*
Specifies the queue of the port. The range is from 0 through 7.

## Modes

Global configuration mode

## Usage Guidelines

Port descriptors set the limit for the ports. The minimum limit for port descriptors is 16. The maximum limit of the port descriptors depends on the hardware device. The minimum limit for queue descriptors is 16. The system default queue descriptors are different for different platforms.

The **no** form of the command deletes the port descriptors.

## Examples

The following example configures the port descriptors.

```
device(config)# qd-descriptor 1 2 76
```

The following example configures the queue descriptors.

```
device(config)# qd-descriptor 1 2 76 2
```

# qos egress-buffer-profile port-share-level

Configures an egress buffer profile for the share port level.

## Syntax

**qos egress-buffer-profile** *user-profile-name* **port-share-level** *level*

**no qos egress-buffer-profile** *user-profile-name* **port-share-level** *level*

## Command Default

The default egress buffer profile level is level4-1/9 for 1/9 of the buffers in buffer memory.

## Parameters

*user-profile-name*

Specifies the name of the egress buffer profile to be configured.

*level*

Specifies the number of buffers that can be used in the buffer memory. The following levels are supported.

| Level | Sharing-pool buffers |
|---|---|
| level3-1/16 | 1/16 of the buffers in buffer memory |
| level4-1/9 | 1/9 of the buffers in buffer memory |
| level5-1/5 | 1/5 of the buffers in buffer memory |
| level6-1/3 | 1/3 of the buffers in buffer memory |
| level7-1/2 | 1/2 of the buffers in buffer memory |
| level8-2/3 | 2/3 of the buffers in buffer memory |

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on the Ruckus ICX 7150.

After creating the profile, you can attach it to one or more ports.

The **no** form of this command resets the egress buffer profile level to its default value of level4-1/9 for 1/9 of the buffers in the buffer memory.

You must use the **no egress-buffer-profile** command to detach a profile from any ports that are using it before you can configure the **no qos egress-buffer-profile** command to delete it.

# Examples

The following example creates an egress buffer profile named egress2 with a maximum of 1/16 of the buffers in buffer memory.

```
device(config)# qos egress-buffer-profile egress2 port-share-level level3-1/16
```

# History

| Release version | Command history |
|---|---|
| 08.0.60 | This command was introduced. |

# qos egress-buffer-profile queue-share-level

Configures an egress buffer profile for the share queue level.

## Syntax

**qos egress-buffer-profile** *user-profile-name* **queue-share-level** *level queue-number*

**no qos egress-buffer-profile** *user-profile-name* **queue-share-level** *level queue-number*

## Command Default

The default share level for an egress buffer profile is:

| Queue | Share level |
|---|---|
| 0 | level4-1/9 |
| 1 | level3-1/16 |
| 2 | level3-1/16 |
| 3 | level3-1/16 |
| 4 | level3-1/16 |
| 5 | level3-1/16 |
| 6 | level3-1/16 |
| 7 | level3-1/16 |

The level4-1/9 share level for queue 0 uses 1/9 of the buffers in the sharing pool. The level3-1/16 share level for queue 1 through 7 uses 1/16 of the buffers in the sharing pool for each queue.

## Parameters

*user-profile-name*
Specifies the name of the egress buffer profile to be configured.

**queue-share-level** *level*
Specifies the number of buffers that can be used in a sharing pool. Eight levels are supported.

*queue-number*
Specifies the queue to apply the buffer limit to. There are eight hardware queues per port.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on Ruckus ICX 7250, ICX 7450, and ICX 7750 devices. This command is not supported on the Ruckus ICX 7150.

The **no** form of this command deletes the egress buffer profile.

You can attach an egress buffer profile to a port.

You must configure the **no egress-buffer-profile** command to detach a profile from any ports that are using it before you can configure the **no qos egress-buffer-profile** command to delete it.

The higher the sharing level, the better the port absorb micro-burst. However, higher-sharing levels of 7 and 8 may compromise QoS functions and create uneven distribution of traffic during periods of congestion.

The following eight queue-share levels are supported:

| Level | Sharing-pool buffers |
|---|---|
| level1-1/64 | 1/64 of buffers in the sharing pool |
| level2-1/32 | 1/32 of buffers in the sharing pool |
| level3-1/16 | 1/16 of buffers in the sharing pool |
| level4-1/9 | 1/9 of buffers in the sharing pool |
| level5-1/5 | 1/5 of buffers in the sharing pool |
| level6-1/3 | 1/3 of buffers in the sharing pool |
| level7-1/2 | 1/2 of buffers in the sharing pool |
| level8-2/3 | 2/3 of buffers in the sharing pool |

# Examples

The following example creates an egress buffer profile named port-40G.

```
Device(config)# qos egress-buffer-profile port-40G queue-share-level
  level1-1/64   1/64 of buffers in the sharing pool
  level2-1/32   1/32 of buffers in the sharing pool
  level3-1/16   1/16 of buffers in the sharing pool
  level4-1/9    1/9 of buffers in the sharing pool
  level5-1/5    1/5 of buffers in the sharing pool
  level6-1/3    1/3 of buffers in the sharing pool
  level7-1/2    1/2 of buffers in the sharing pool
  level8-2/3    2/3 buffers in the sharing pool
```

The following example configures queue 0 on the egress buffer profile named port-40G to use 1/5 of sharing pool.

```
Device(config)# qos egress-buffer-profile port-40G port-40G queue-share-level level5-1/5 0
```

The following example configures queue 1 on the egress buffer profile named port-40G to use 1/64 of the sharing pool.

```
Device(config)# qos egress-buffer-profile port-40G port-40G queue-share-level level1-1/64 1
```

The following example attaches the egress buffer profile named port-40G to ports 1/2/1 to 1/2/6.

```
Device(config)# interface ethernet 1/2/1 to 1/2/6
Device(config-mif-1/2/1-1/2/6)#egress-buffer-profile port-40G
Device(config-mif-1/2/1-1/2/6)#end
```

The following example shows the error if you try to delete a profile that is attached to a port.

```
Device(config)# no qos egress-buffer-profile port-40G
Error - Egress Profile port-40G is active on Port 1/2/1. It must be deactivated from port before
deleting.
```

The following example detaches the egress buffer profile named port-40G from ports 1/2/1 to 1/2/6 and then delete the profile.

```
Device(config)# interface ethernet 1/2/1 to 1/2/6
Device(config-mif-1/2/1-1/2/6)# no egress-buffer-profile port-40G
Device(config-mif-1/2/1-1/2/6)#exit
Device(config)# no qos egress-buffer-profile port-40G
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |
| 8.0.60 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7150. |

# qos egress-shape-ifg-bytes

Configures egress shaper IFG bytes.

## Syntax

**qos egress-shape-ifg-bytes** *value-in-bytes*

**no qos egress-shape-ifg-bytes** *value-in-bytes*

## Command Default

By default, a value of 20 bytes is configured.

## Parameters

*value-in-bytes*

Specifies the number of preamble and IFG bytes to be added to egress shaping in the range 1 through 127.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported on ICX 7150, ICX 7250, ICX 7450, ICX 7650, and ICX 7750 devices.

For the ICX 7650, we recommend using a value greater than or equal to 20 bytes.

The **no** form of the command restores the default value of 20.

## Examples

The following example configures an egress shaper IFG bytes value of 25.

```
Device(config)# qos egress-shaper-ifg-bytes 25
```

The following example restores the default egress shaper IFG bytes value of 20.

```
Device(config)# no qos egress-shaper-ifg-bytes 25
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | Added a usage guideline concerning the support of this command on the Ruckus ICX 7650. |

# qos ingress-buffer-profile

Configures an ingress buffer profile.

## Syntax

**qos ingress-buffer-profile** *user-profile-name* **priority-group** *priority-group-number* **xoff** *shared-level*

**no qos ingress-buffer-profile** *user-profile-name* **priority-group** *priority-group-number* **xoff** *shared-level*

## Command Default

An ingress buffer profile is not configured.

## Parameters

*user-profile-name*
Specifies the name of the ingress buffer profile to be configured.

**priority-group** *priority-group-number*
Specifies the priority group (PG) number with the XOFF threshold level that must be configured.

**xoff** *shared-level*
Specifies the per-PG buffer threshold to trigger sending of priority flow control (PFC).

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on Ruckus ICX 7250, ICX 7450, and ICX 7750 devices. This command is not supported on the Ruckus ICX 7150 or the Ruckus ICX 7650.

You can attach an ingress buffer profile to a port.

You must configure the **no ingress-buffer-profile** command to detach a profile from any ports that are using it before you can configure the **no qos ingress-buffer-profile** command to delete it.

The higher the sharing level, the better the port absorbs micro-bursts before reaching the XOFF threshold limit.

If PFC is enabled on a PG, and per-port with a user-defined ingress buffer profile attached to a port, the port maximum XOFF threshold level is 50 percent of service pool 1. The port maximum is used as a cap to prevent a port from using too many buffers. Under normal conditions, the PG XOFF limit is reached first.

If a PG is not enabled to send globally, any XOFF value configured has no effect.

The default ingress buffer profiles are as follows:

- For PFC disabled ports, the default PG XOFF limit is level7-1/2.
- For PFC enabled ports, the default PG XOFF limit is level2-1/32.

The following six PG XOFF limits are supported:

| Level | Sharing-pool buffers |
|---|---|
| level1-1/64 | 1/64 of buffers in the sharing pool |
| level2-1/32 | 1/32 of buffers in the sharing pool |
| level3-1/16 | 1/16 of buffers in the sharing pool |
| level4-1/9 | 1/9 of buffers in the sharing pool |
| level5-1/5 | 1/5 of buffers in the sharing pool |
| level6-1/3 | 1/3 of buffers in the sharing pool |
| level7-1/2 | 1/2 of buffers in the sharing pool |

The **no** form of the command deletes the ingress buffer profile.

# Examples

The following example creates an ingress buffer profile for PG 0 with a PG XOFF limit of 1/3 of buffers in the sharing pool.

```
Device(config)# qos ingress-buffer-profile ing1 priority-group 0 xoff level6-1/3
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.60 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7150. |
| 08.0.70 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7650. |

# qos mechanism

Configures the Quality of Service (QoS) queuing method.

## Syntax

**qos mechanism** { **strict** | **weighted** | **mixed-sp-wrr** }

**no qos mechanism** { **strict** | **weighted** | **mixed-sp-wrr** }

## Command Default

By default, devices use the Weighted Round Robin (WRR) method of packet prioritization. WRR ensures that all queues are serviced during each cycle

## Parameters

**strict**

Changes the method to strict order scheduling (Strict Priority (SP)).

**weighted**

Changes the method to weighted scheduling (WRR).

**mixed-sp-wrr**

Changes the method to both strict scheduling and weighted scheduling.

## Modes

Global configuration mode

## Usage Guidelines

By default, when you select the combined Strict Priority (SP) and WRR queueing method, the device assigns strict priority to traffic in qosp6 and qosp7 and weighted round robin priority to traffic in qosp0 through qosp5.

The **no** form of the command configures the device to use the WRR method of packet prioritization.

## Examples

The following example shows changes the method to strict priority scheduling.

```
device(config)# qos mechanism strict
```

# qos monitor-queue-drop-counters

Configures the port that the Ruckus ICX 7150 device monitors for the incrementing of the egress queue drop counters.

## Syntax

**qos monitor-queue-drop-counters** *port-id*

**no qos monitor-queue-drop-counters**

## Command Default

By default, the egress queue drop counters is associated with the local CPU port that the device monitors for control packet drops.

## Parameters

*port-id*

Specifies the port ID to associate with the egress queue drop counters.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on the Ruckus ICX 7150.

The device has one set of queue drop counters that must be associated to a port. Only one port in a device can be monitored.

Use this command when traffic loss occurs on a port and you want to verify if the queue drop counters increment.

The **no** form of the command reset the monitoring to the internal local CPU port.

## Examples

The following example configures port 1/1/12 for monitoring on the egress queue drop counters.

```
device(config)# qos monitor-queue-drop-counters 1/1/12
```

## History

| Release version | Command history |
|---|---|
| 08.0.60 | This command was introduced. |

# qos name

Renames the queue.

## Syntax

**qos name** *old-name new-name*

## Command Default

The default queue names are qosp7, qosp6, qosp5, qosp4, qosp3, qosp2, qosp1, and qosp0.

## Parameters

*old-name*

Specifies the name of the queue before the change.

*new-name*

Specifies the new name of the queue. The name can be an alphanumeric string up to 32 characters long.

## Modes

Global configuration mode

## Examples

The following example renames the queue qosp3 to 92-octane.

```
device(config)# qos name qosp3 92-octane
```

# qos priority-to-pg

Configures priority-to-priority-group (PG) mapping for priority flow control (PFC).

## Syntax

**qos priority-to-pg qosp0** *priority-PG-map* **qosp1** *priority-PG-map* **qosp2** *priority-PG-map* **qosp3** *priority-PG-map* **qosp4** *priority-PG-map* **qosp5** *priority-PG-map* **qosp6** *priority-PG-map* **qosp7** *priority-PG-map*

**no qos priority-to-pg**

## Command Default

Priority-to-PG mapping is not configured.

## Parameters

**qosp0** through **qosp7**
Configures the internal priority based on classification in the range 0 through 7.

*priority-PG-map*
Specifies the internal priority-to-PG mapping. The range is from 0 through 3.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported only on Ruckus ICX 7250, ICX 7450, and ICX 7750 devices. This command is not supported on the Ruckus ICX 7150 or the Ruckus ICX 7650.

You must configure the **priority-flow-control enable** command to enable PFC globally before you configure priority-to-PG mapping.

**NOTE**
Default mapping, mapping priorities, and mapping restrictions changed in FastIron Release 08.0.20. The following restrictions apply:

- Priority 7, and only Priority 7, is always mapped to PG4.
- PG4 is always lossy.
- PFC cannot be enabled on PG4.
- Priorities 0 through 5 can be mapped to PG0, PG1, and PG2. They cannot be mapped to PG3 or PG4.

The default values of priority-to-PG maps:

- QoS internal priority 0 is mapped to PG 0
- QoS internal priority 1 is mapped to PG 0
- QoS internal priority 2 is mapped to PG 1

- QoS internal priority 3 is mapped to PG 1
- QoS internal priority 4 is mapped to PG 1
- QoS internal priority 5 is mapped to PG 2
- QoS internal priority 6 is mapped to PG 2
- QoS internal priority 7 is mapped to PG 4

The default values of priority-to-PG maps in releases prior to Release 08.0.20:

- QoS internal priority 0 is mapped to PG 0
- QoS internal priority 1 is mapped to PG 0
- QoS internal priority 2 is mapped to PG 1
- QoS internal priority 3 is mapped to PG 1
- QoS internal priority 4 is mapped to PG 1
- QoS internal priority 5 is mapped to PG 2
- QoS internal priority 6 is mapped to PG 2
- QoS internal priority 7 is mapped to PG 2

In releases prior to Release 08.0.20, you can map QoS internal priority 7 to PG 3. You can also map any other priority to PG 3 if it meets the following requirements:

- Lower priorities are mapped to lower PGs.
- PGs are configured in ascending order.
- Multiple priorities in a single PG must be consecutive.

Priority-to-PG mapping is not configurable in other modes. Symmetrical and asymmetrical 802.3x flow control modes have their own default priority-to-PG mapping.

You must configure PGs in ascending order, 0 to 3. You can configure a higher-order PG only if all the lower-order PGs have some mapped priorities.

The **no** form of the command restores the default priority-to-PG map.

# Examples

The following example configures a priority-to-PG map.

```
Device(config)# priority-flow-control enable
Device(config)# qos priority-to-pg qosp0 0 qosp1 1 qosp2 1 qosp3 1 qosp4 2 qosp5 2 qosp6 2 qosp7 4
```

The following example restores the default priority-to-PG map.

```
Device(config)# no qos priority-to-pg qosp0 0 qosp1 1 qosp2 1 qosp3 1 qosp4 2 qosp5 2 qosp6 2 qosp7 4
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |
| 08.0.20 | This command was modified to change priority 7-to-PG4 mapping and mapping restrictions for priorities 0 through 5. |
| 08.0.60 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7150. |

| Release version | Command history |
|---|---|
| 08.0.70 | Added a usage guideline concerning the support of this command and that this command is not supported on the Ruckus ICX 7650. |

# qos profile

Changes the minimum bandwidth percentages of the eight Weighted Round Robin (WRR) queues.

## Syntax

**qos profile** { *name7* { **sp** | *percentage* } ... *name0* { **sp** | *percentage* }

**no qos profile** { *name7* { **sp** | *percentage* } ... *name0* { **sp** | *percentage* }

## Command Default

The eight QoS queues on FastIron devices receive the minimum guaranteed percentages of a port's total bandwidth, as shown in the following table. Note that the defaults differ when jumbo frames are enabled.

## Parameters

*name*

Specifies the name of a queue. You can specify the queues in any order on the command line, *but you must specify each queue*.

**sp**

Changes the method to strict priority scheduling.

*percentage*

Specifies the percentage of the device outbound bandwidth that is allocated to the queue. QoS queues require a minimum bandwidth percentage of 3 percent for each priority. When jumbo frames are enabled, the minimum bandwidth requirement is 8 percent. If these minimum values are not met, QoS may not be accurate.

## Modes

Global configuration mode

## Usage Guidelines

When the queuing method is WRR, the software internally translates the percentages into weights. The weight associated with each queue controls how many packets are processed at a given stage through the weighted round robin algorithm.

**TABLE 10 Default minimum bandwidth percentages**

| Queue | Default minimum percentage of bandwidth | |
|---|---|---|
| | Without jumbo frames | With jumbo frames |
| qosp7 | 75% | 44% |
| qosp6 | 7% | 8% |
| qosp5 | 3% | 8% |
| qosp4 | 3% | 8% |
| qosp3 | 3% | 8% |
| qosp2 | 3% | 8% |

**TABLE 10 Default minimum bandwidth percentages (continued)**

| Queue | Default minimum percentage of bandwidth | |
|---|---|---|
| qosp1 | 3% | 8% |
| qosp0 | 3% | 8% |

The **no** form of the command restores the default bandwidth percentages.

# Examples

The following example changes the bandwidth percentages for the queues.

```
device(config)# qos profile qosp7 25 qosp6 15 qosp5 12 qosp4 12 qosp3 10 qosp2
10 qosp1 10 qosp0 6

Profile qosp7 : Priority7 bandwidth requested 25% calculated 25%
Profile qosp6 : Priority6 bandwidth requested 15% calculated 15%
Profile qosp5 : Priority5 bandwidth requested 12% calculated 12%
Profile qosp4 : Priority4 bandwidth requested 12% calculated 12%
Profile qosp3 : Priority3 bandwidth requested 10% calculated 10%
Profile qosp2 : Priority2 bandwidth requested 10% calculated 10%
Profile qosp1 : Priority1 bandwidth requested 10% calculated 10%
Profile qosp0 : Priority0 bandwidth requested 6% calculated 6%
```

# qos scheduler-profile

Configures a user-defined Quality of Service (QoS) scheduler profile.

## Syntax

**qos scheduler-profile** *user-profile-name* { **mechanism** *scheduling-mechanism* | **profile** [ **qosp0** *wt0* | **qosp1** *wt1* | **qosp2** *wt2* | **qosp3** *wt3* | **qosp4** *wt4* | **qosp5** *wt5* | **qosp6** *wt6* | **qosp7** *wt7* ] }

**no qos scheduler-profile** *user-profile-name*

## Command Default

A user-defined QoS scheduler profile is not configured.

## Parameters

*user-profile-name*

Specifies the name of the scheduler profile to be configured.

**mechanism** *scheduling-mechanism*

Configures the queue assignment with the specified scheduling mechanism. The following scheduling mechanisms are supported:

**mixed-sp-wrr**

Specifies mixed strict-priority (SP) and weighted scheduling.

**strict**

Specifies SP scheduling.

**weighted**

Specifies weighted scheduling.

**profile qosp0-7**

Configures the profile based on classification in the range 0 through 7.

**wt0-7**

Specifies the bandwidth percentage for the corresponding QoS profile. The range is from 0 through 7.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command removes the scheduler profile configuration.

You can use the **scheduler-profile** command to attach a user scheduler profile to a port. If you want to remove a scheduler-profile you must ensure that it is not attached to any port.

On ICX 7750 and ICX 7450 devices, changing the global scheduler and port scheduler on running traffic may cause traffic loss.

The default QoS-profile weights for each queue using a weighted QoS mechanism are as follows:

| Profile | Priority | Weighted bandwidth |
|---|---|---|
| Profile qosp7 | Priority7(Highest) | Bandwidth requested 44% calculated 44% |
| Profile qosp6 | Priority6 | Bandwidth requested 8% calculated 8% |
| Profile qosp5 | Priority5 | Bandwidth requested 8% calculated 8% |
| Profile qosp4 | Priority4 | Bandwidth requested 8% calculated 8% |
| Profile qosp3 | Priority3 | Bandwidth requested 8% calculated 8% |
| Profile qos2 | Priority2 | Bandwidth requested 8% calculated 8% |
| Profile qosp1 | Priority1 | Bandwidth requested 8% calculated 8% |
| Profile qosp0 | Priority0 (Lowest) | Bandwidth requested 8% calculated 8% |

| Per-queue details | Bandwidth percentage |
|---|---|
| Class 0 | 3 |
| Class 1 | 3 |
| Class 2 | 3 |
| Class 3 | 3 |
| Class 4 | 3 |
| Class 5 | 3 |
| Class 6 | 7 |
| Class 7 | 75 |

The default QoS-profile weights for each queue using a mixed QoS mechanism are as follows:

| Per-queue details | Bandwidth percentage |
|---|---|
| Class 0 | 15 |
| Class 1 | 15 |
| Class 2 | 15 |
| Class 3 | 15 |
| Class 4 | 15 |
| Class 5 | 25 |
| Class 6 | sp |
| Class 7 | sp |

The total weight (wt0-wt7) in both weighted and mixed mechanism must be 100 percent.

The minimum value for any weight is 1.

A maximum of eight scheduler profiles are supported.

# Examples

The following example configures a QoS scheduler profile named user1, with weighted scheduling, and specify the bandwidth percentage for each QoS class:.

```
Device(config)# qos scheduler-profile user1 mechanism weighted
Device(config)# qos scheduler-profile user1 profile qosp0 1 qosp1 1 qosp2 10 qosp3 10 qosp4 10 qosp5 10
qosp6 20 qosp7 38
```

The following example configures a QoS scheduler profile named user2, with SP scheduling.

```
Device(config)# qos scheduler-profile user2 mechanism strict
```

The following example configures a QoS scheduler profile named user3, with mixed SP and weighted scheduling.

```
Device(config)# qos scheduler-profile user3 mechanism mixed-sp-wrr
```

The following example removes a QoS scheduler profile named user3.

```
Device(config)# no qos scheduler-profile user3
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# qos sflow-set-cpu-rate-limit

Sets the CPU rate limit for sFlow.

## Syntax

**qos sflow-set-cpu-rate-limit** *packet-rate burst-size*

**no qos sflow-set-cpu-rate-limit** *packet-rate burst-size*

## Command Default

A CPU rate limit for sFlow is configured with the default values of 100 sFlow sampled packets per second (PPS) and a burst size of 5000 B.

## Parameters

*packet-rate*

Specifies the number of sFlow sampled PPS into the CPU. The value is measured in PPS and ranges from 1 to 1000.

*burst-size*

Specifies the burst size. The value is measured in bytes and ranges from 1 to 99999.

## Modes

Global configuration mode

## Usage Guidelines

If the burst size is set low more packets are subject to rate limiting. If the burst size is set too high fewer packets are subject to rate limiting.

You should not set the burst size less than 10 times the maximum transmission unit of the traffic.

The recommended settings are 1000 PPS with a maximum burst size of 5000 B.

The **no** form of this command returns the device to the default CPU rate limit for sFlow.

## Examples

The following example uses the recommended settings to configure the CPU rate limit for sFlow.

```
device(config)# qos sflow-set-cpu-rate-limit 1000 5000
```

To view the CPU rate limit for sFlow use the following command.

```
device(config)# show qos sflow-rate-limit
Queue-Num       Rate-Limt       Burst-Size
Queue13           1000            5000
device(config)#
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# qos tagged-priority

Changes the VLAN priority of 802.1p to hardware forwarding queue mappings.

## Syntax

**qos tagged-priority** *num queue*

**no qos tagged-priority** *num queue*

## Parameters

*num*

Specifies the VLAN priority. The value can range from 0 to 7.

*queue*

Specifies the hardware forwarding queue on which you are reassigning the priority. The default queue names are as follows: qosp7, qosp6, qosp5, qosp4, qosp3, qosp2, qosp1, qosp0.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the VLAN priority to 802.1p.

## Examples

The following example maps VLAN priority 2 to hardware forwarding queue qosp0.

```
device(config)# qos tagged-priority 2 qosp0
```

# qos-internal-trunk-queue

Modifies the dynamic buffer-share level of inter-packet-processor (inter-pp) HiGig links egress queues on ICX 7450 devices.

## Syntax

**qos-internal-trunk-queue** *level queue*

**no qos-internal-trunk-queue** *level queue*

## Command Default

The buffer share level defaults are:

| Queue | Share level |
|---|---|
| 0 | level4-1/9 |
| 1 | level3-1/16 |
| 2 | level3-1/16 |
| 3 | level3-1/16 |
| 4 | level3-1/16 |
| 5 | level3-1/16 |
| 6 | level3-1/16 |
| 7 | level3-1/16 |

## Parameters

*level*

Specifies the number of buffers that can be used in a sharing pool. ICX 7450 devices support eight levels.

*queue*

Specifies the queue to apply the buffer limit to. Each port has eight hardware queues.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default queue share level on the specified queue.

**NOTE**
This command is supported only on ICX 7450 devices or across stack units or for ports across master and slave packet-processor (pp) devices in ICX7450-48 units.

The following eight queue-share levels are supported:

| Level | Sharing-pool buffers |
|---|---|
| level1-1/64 | 1/64 of buffers in the sharing pool |
| level2-1/32 | 1/32 of buffers in the sharing pool |
| level3-1/16 | 1/16 of buffers in the sharing pool |
| level4-1/9 | 1/9 of buffers in the sharing pool |
| level5-1/5 | 1/5 of buffers in the sharing pool |
| level6-1/3 | 1/3 of buffers in the sharing pool |
| level7-1/2 | 1/2 of buffers in the sharing pool |
| level8-2/3 | 2/3 of buffers in the sharing pool |

# Examples

The following example configures the buffer share level of inter-packet-processor (inter-pp) HiGig links egress queues.

```
ICX7450-48P Router(config)#qos-internal-trunk-queue
  level1-1/64   1/64 of buffers in the sharing pool
  level2-1/32   1/32 of buffers in the sharing pool
  level3-1/16   1/16 of buffers in the sharing pool
  level4-1/9    1/9 of buffers in the sharing pool
  level5-1/5    1/5 of buffers in the sharing pool
  level6-1/3    1/3 of buffers in the sharing pool
  level7-1/2    1/2 of buffers in the sharing pool
  level8-2/3    2/3 buffers in the sharing pool
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# qos-tos map dscp-priority

Changes the ifferentiated Services Code Point (DSCP)-to-internal-forwarding-priority mappings.

## Syntax

**qos-tos map dscp-priority** *dscp-value1* [ ...*dscp-value8* ] **to** *priority*

**no qos-tos map dscp-priority** *dscp-value1* [ ...*dscp-value8* ] **to** *priority*

## Command Default

Refer the Usage Guidelines.

## Parameters

*dscp-value*

Specifies the DSCP value ranges that you are remapping. You can map up to eight DSCP values to the same forwarding priority in the same command.

**to**

Configures the DSCP value to the new internal forwarding priority.

*priority*

Specifies the internal forwarding priority.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command restores the default value.

**TABLE 11 Default DSCP-to-internal-forwarding-priority mappings**

| Internal forwarding priority | DSCP value |
|---|---|
| 0 [lowest priority queue] | 0–7 |
| 1 | 8–15 |
| 2 | 16–23 |
| 3 | 24–31 |
| 4 | 32–39 |
| 5 | 40–47 |
| 6 | 48–55 |
| 7 [highest priority queue] | 56–63 |

DSCP values range from 0 through 63, whereas the internal forwarding priority values range from 0 through 7. Any DSCP value within a given range is mapped to the same internal forwarding priority value. For example, any DSCP value from 8 through 15 maps to priority 1.

# Examples

The following example changes the DSCP-to-internal-forwarding-priority mappings.

```
device(config)# qos-tos map dscp-priority 0 2 3 4 to 1
```

# radius-client coa host

Configures the key to be used between the Change of Authorization (CoA) client and FastIron device.

## Syntax

**radius-client coa host** { *addr* | *name* } [ **key** *key-string* ]

**no radius-client coa host** { *addr* | *name* } [ **key** *key-string* ]

## Command Default

No key is configured between the CoA client and device.

## Parameters

*addr*
Address of the CoA host.

*name*
Name of the CoA host.

**key** *key-string*
The key required to be used between the CoA client and FastIron device.

## Modes

Global configuration mode

## Usage Guidelines

**no**

RADIUS Change of Authorization (CoA) messages from clients configured through this command will be processed. CoA messages from unconfigured clients will be discarded.

## Examples

The following example displays the configuration between CoA host and the device.

```
device(config)# radius-client coa host 10.21.240.46 key 0 Foundry1#
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# radius-client coa port

Changes the default CoA (Change of Authorization) port number.

## Syntax

**radius-client coa port** *udp-port-number*

**no radius-client coa port** *udp-port-number*

## Command Default

The CoA port number is 3799.

## Parameters

*udp-port-number*
The number of the UDP port.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command restores the default port number (3799).

## Examples

The following example changes the CoA port number to 3000.

```
device(config)# radius-client coa port 3000
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# radius-server accounting

Configures to send interim updates of accounting messages to the RADIUS server at regular intervals..

## Syntax

**radius-server accounting** { **interim-updates** | **interim-interval** *value*}

**no radius-server accounting** { **interim-updates** | **interim-interval** *value*}

## Command Default

Accounting updates are disabled. The default interim interval is zero.

## Parameters

**interim-updates**
Enables the interim accounting updates.

**interim-interval** *value*
Sets the interval between each interim update. Default value is 0. The range of valid values is from 5 through 1440 minutes.

## Modes

Global configuration mode

## Usage Guidelines

The RADIUS accounting for 802.1X authentication and MAC authentication accepts either the interim update interval value configured using the RADIUS attribute or the interval time value set on the device, whichever is higher.

The **no** form of the command resets the feature to the default values.

## Examples

The following example enables interim updates and set accounting update intervals for 802.1X authentication accounting and MAC authentication accounting as 10 minutes.

```
device# configure terminal
device(config)# radius-server accounting interim-updates
device(config)# radius-server accounting interim-interval 10
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# radius-server dead-time

Configures the interval at which the test user message is sent to the server to check the status of non-responding servers that are marked as dead.

## Syntax

**radius-server dead-time** *time*

**no radius-server dead-time** *time*

## Command Default

RADIUS dead time is not enabled.

## Parameters

*time*

The time interval between successive server requests to check the availability of the RADIUS server in minutes. The valid values are from 1 through 5 minutes.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the dead time interval.

## Examples

The following example configures the RADIUS server dead time as four minutes.

```
device(config)# radius-server deadtime 4
```

# radius-server enable

Configures the device to allow RADIUS server management access only to clients connected to ports within the port-based VLAN.

## Syntax

**radius-server enable vlan** *vlan-number*

**no radius-server enable vlan** *vlan-number*

## Command Default

By default, access is allowed on all ports.

## Parameters

**vlan** *vlan-number*

Configures access only to clients connected to ports within the VLAN.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the restriction.

You can restrict management access to a Ruckus device to ports within a specific port-based VLAN. VLAN-based access control works in conjunction with other access control methods. Clients connected to ports that are not in the VLAN are denied management access.

## Examples

The following example shows how to allow RADIUS server access only to clients in a specific VLAN.

```
device(config)# radius-server enable vlan 10
```

# radius-server host

Configures the Remote Authentication Dial-In User Service (RADIUS) server.

## Syntax

**radius-server host** { *ipv4-address* | *host-name* | *ipv6-address* } [ **auth-port** *port-num* [ **acct-port** *port-num* [ { **accounting-only** | **authentication-only** | **default** } [ **ssl-auth-port** *port-num* [ **accounting-only** | **authentication-only** | **default** ] [ **key** *key-string* [ **dot1x** | **mac-auth** | **no-login** | **web-auth** ] [ **port-only** ] ] ] ] ]

**no radius-server host** { *ipv4-address* | *host-name* | *ipv6-address* } [ **auth-port** *port-num* [ **acct-port** *port-num* [ { **accounting-only** | **authentication-only** | **default** } [ **ssl-auth-port** *port-num* [ **accounting-only** | **authentication-only** | **default** ] [ **key** *key-string* [ **dot1x** | **mac-auth** | **no-login** | **web-auth** ] [ **port-only** ] ] ] ] ]

## Command Default

The RADIUS server host is not configured.

## Parameters

*ipv4-address*
Configures the IPv4 address of the RADIUS server.

*host-name*
Configures the host name of the RADIUS server.

*ipv6-address*
Configures the IPv6 address of the RADIUS server.

**auth-port** *port-num*
Configures the authentication UDP port. The default value is 1812.

**acct-port** *port-num*
Configures the accounting UDP port. The default value is 1813.

**accounting-only**
Configures the server to be used only for accounting.

**authentication-only**
Configures the server to be used only for authentication.

**default**
Configures the server to be used for any AAA operation.

**ssl-auth-port** *port-num*
Specifies that the server is a RADIUS server running over a TLS-encrypted TCP session. Only one of **auth-port** or **ssl-auth-port** can be specified. If neither is specified, it defaults to the existing default behavior, which uses the default auth-port of 1812 and 1813 for accounting with no TLS encryption. The default destination port number for RADIUS over TLS is TCP/2083.There are no separate ports for authentication, accounting, and dynamic authorization changes. The source port is arbitrary. TLS-encrypted sessions support both IPv4 and IPv6.

**accounting-only**
Configures the server to be used only for accounting.

**authentication-only**
Configures the server to be used only for authentication.

**default**
Configures the server to be used for any AAA operation.

**key** *key-string*
Configures the RADIUS key for the server.

**dot1x**
Configures support for EAP for 802.1X authentication.

**mac-auth**
Configures the server to be used only for MAC authentication.

**no-login**
Configures the server not to be used for Telnet, SSH, console, EXEC, or Web-management AAA.

**web-auth**
Configures the server to be used only for Web authentication.

**port-only**
Specifies that the server will be used only to authenticate users on ports to which it is mapped.

# Modes

Global configuration mode

# Usage Guidelines

Use this command to identify a RADIUS server to authenticate access to a Ruckus device. You can specify up to eight servers. If you add multiple RADIUS authentication servers to the Ruckus device, the device tries to reach them in the order you add them. To use a RADIUS server to authenticate access to a Ruckus device, you must identify the server to the Ruckus device. In a RADIUS configuration, you can designate a server to handle a specific AAA task. For example, you can designate one RADIUS server to handle authorization and another RADIUS server to handle accounting. You can specify individual servers for authentication and accounting, but not for authorization. You can set the RADIUS key for each server.

TLS-encrypted TCP sessions are not supported by management VRF.

The **no** form of the command removes the RADIUS sever host configuration.

# Examples

The following example shows how to configure a RADIUS server to authenticate access to a Ruckus device.

```
device(config)# radius-server host 192.168.10.1
```

The following example configures non-default UDP ports for authorization and accounting.

```
device(config)# radius-server host 1.2.3.4 auth-port 100 acct-port 200
device(config)# show aaa
***** TACACS server not configured
Radius default key: ...
Radius retries: 3
Radius timeout: 3 seconds
Radius Server:         IP=172.26.67.12 SSL Port=2083 Usage=any
                    Key=...
                    opens=0 closes=0 timeouts=0 errors=0
                    packets in=0 packets out=0
                    IPv4 Radius Source address: IP=0.0.0.0              IPv6 Radius Source
Address:           IP=::
Radius Server:         IP=1.2.3.4 Auth Port=100 Acct Port=200 Usage=any
                    Key=...
                    opens=0 closes=0 timeouts=0 errors=0
                    packets in=0 packets out=0
                    IPv4 Radius Source address: IP=0.0.0.0              IPv6 Radius Source
Address:           IP=::
```

The following example shows how to specify different RADIUS servers for authentication and accounting.

```
device(config)# radius-server host 10.2.3.4 auth-port 1800 acct-port 1850 default key abc
device(config)# radius-server host 10.2.3.5 auth-port 1800 acct-port 1850 authentication-only key def
device(config)# radius-server host 10.2.3.6 auth-port 1800 acct-port 1850 accounting-only key ghi
```

The following example shows how to map the 802.1X port to a RADIUS server.

```
device(config)# radius-server host 10.2.3.4 auth-port 1800 acct-port 1850 default key abc dot1x
```

The following example shows how to configure a RADIUS server for TLS support.

```
device(config)# radius-server host 172.26.67.12 ssl-auth-port 2083 default key whatever
device(config)# show aaa
***** TACACS server not configured
Radius default key: ...
Radius retries: 3
Radius timeout: 3 seconds
Radius Server:         IP=172.26.67.12 SSL Port=2083 Usage=any
                    Key=...
                    opens=0 closes=0 timeouts=0 errors=0
                    packets in=0 packets out=0
                    IPv4 Radius Source address: IP=0.0.0.0              IPv6 Radius Source
Address:           IP=::
```

The following example configures the RADIUS server to be used for both MAC authentication and login features.

```
device# configure terminal
device(config)# radius-server host 10.26.67.13 auth-port 1812 acct-port 1813 default key ruckus mac-auth
```

The following example configures the RADIUS server to be used for login features and flexible authentication.

```
device# configure terminal
device(config)# radius-server host 10.26.67.13 auth-port 1812 acct-port 1813 default key ruckus
```

The following example excludes the RADIUS server for all login features.

```
device# configure terminal
device(config)# radius-server host 10.26.67.13 auth-port 1812 acct-port 1813 default key ruckus no-login
```

The following example uses the RADIUS server for flexible authentication modules.

```
device# configure terminal
device(config)# radius-server host 10.26.67.13 auth-port 1812 acct-port 1813 default key ruckus mac-
auth dot1x no-login
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was updated with **mac-auth** and **web-auth** options. |
| 08.0.80 | This command was modified to add the **no-login** option. |

# radius-server key

Configures the value that the device sends to the RADIUS server when trying to authenticate user access.

## Syntax

**radius-server key** *key-string*

**no radius-server key** *key-string*

## Command Default

The RADIUS server key is not configured.

## Parameters

*key-string*

Specifies the key as an ASCII string. The value for the key parameter on the device should match the one configured on the RADIUS server. The key can be from 1 through 64 characters in length and cannot include any space characters.

## Modes

Global configuration mode

## Usage Guidelines

The **radius-server key** command is used to encrypt RADIUS packets before they are sent over the network.

The **no** form of the command removes the RADIUS server key configuration.

## Examples

The following example shows how to configure a RADIUS server key.

```
device(config)# radius-server key abc
```

# radius-server retransmit

Configures the maximum number of retransmission attempts for a request when a RADIUS authentication request times out.

## Syntax

**radius-server retransmit** *number*

**no radius-server retransmit** *number*

## Command Default

The default retransmit number is three retries.

## Parameters

*number*

The maximum number of retries the Ruckus software retransmits the request. The valid values are from 1 through 5. The default is 3.

## Modes

Global configuration mode

## Usage Guidelines

When an authentication request times out, the Ruckus software retransmits the request up to the maximum number of retransmission tries configured.

The **no** form of the command removes the configuration.

## Examples

The following example shows how to set the retransmission number to 4.

```
device(config)# radius-server retransmission 4
```

# radius-server test

Sets the user name to be used in the RADIUS request packets for RADIUS dead server detection.

## Syntax

**radius-server test** *user-name*

**no radius-server test**

## Command Default

There is no user name configured.

## Parameters

*user-name*

The false user name used in the server test.

## Modes

Global configuration mode

## Usage Guidelines

The username should not be configured on the server, so that the server responds with Access-Reject message if the server is available.

If the device does not receive a response from a RADIUS server within a specified time limit and number of retries, the RADIUS server is marked as dead. The time limit and number of retries can be manually configured using the **radius-server timeout** and **radius-server retransmit** commands respectively.

The **no** form of the command disables the configuration to send RADIUS request packets with false usernames for RADIUS dead server detection.

## Examples

The following example configures the user name as 'test-user' to test the availability of the server.

```
device# configure terminal
device(config)# radius-server test test-user
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# radius-server timeout

Configures the number of seconds the device waits for a response from a RADIUS server before either retrying the authentication request, or determining that the RADIUS servers are unavailable and moving on to the next authentication method in the authentication method list.

## Syntax

**radius-server timeout** *time*

**no radius-server timeout** *time*

## Command Default

The default timeout value is 3 seconds.

## Parameters

*time*

The timeout value in seconds. Valid values are from 1 through 15 seconds. The default is 3 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the timeout value to the default.

## Examples

The following example shows how to set the RADIUS server timeout value to 10 seconds.

```
device(config)# radius-server timeout 10
```

# raguard

Configures the current interface as a trusted, untrusted, or host Router Advertisement (RA) guard port.

## Syntax

**raguard** { **trust** | **untrust** | **host** }

**no raguard** { **trust** | **untrust** | **host** }

## Parameters

**trust**
Configures an interface as a trusted RA guard port.

**untrust**
Configures an interface as an untrusted RA guard port.

**host**
Configures an interface as a host RA guard port.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command removes the current trusted or untrusted configuration.

A trusted RA guard port forwards all the receive RA packets without inspecting. An untrusted port inspects the received RAs against the RA guard policy's whitelist, prefix list and preference maximum settings before forwarding the RA packets. If an RA guard policy is not configured on an untrusted or host port, all the RA packets are forwarded.

## Examples

The following example configures an interface as a trusted RA guard port:

```
device(config)# interface ethernet1/1/1
device(config-int-e1000-1/1/1)# raguard trust
```

The following example configures an interface as an untrusted RA guard port:

```
device(config)# interface ethernet1/2/1
device(config-int-e1000-1/2/1)# raguard untrust
```

The following example configures an interface as a host RA guard port:

```
device(config)# interface ethernet3/2/1
device(config-int-e1000-3/2/1)# raguard host
```

# rarp

Assigns a static IP RARP entry for static routes.

## Syntax

**rarp** *index mac-address ip-address*

**no rarp** *index mac-address ip-address*

## Command Default

RARP entry is not configured.

## Parameters

*index*

Specifies the static IP RARP entry's index. You can specify an unused number from 1 to the maximum number of RARP entries supported on the device.

*mac-address*

Specifies the static IP RARP entry's MAC address.

*ip-address*

Specifies the static IP RARP entry's IP address.

## Modes

Global configuration mode

## Usage Guidelines

You must configure the RARP entries for the RARP table. The Layer 3 switch can send an IP address in reply to a client RARP request only if create a RARP entry for that client.

The **no** form of the command removes the static IP RARP entry.

## Examples

The following example creates a RARP entry for a client with MAC address 0000.0054.2348. When the Layer 3 switch receives a RARP request from this client, the Layer 3 switch replies to the request by sending IP address 192.53.4.2 to the client.

```
device(config)# rarp 1 0000.0054.2348 192.53.4.2
```

# rate-limit-arp

Limits the number of ARP packets the Ruckus device accepts during each second.

## Syntax

**rate-limit-arp** *number*

**no rate-limit-arp** *number*

## Command Default

ARP rate limiting is not enabled.

## Parameters

*number*

Specifies the number of ARP packets and can be from 0 through 100. If you specify 0, the device will not accept any ARP packets.

## Modes

Global configuration mode

## Usage Guidelines

To prevent the CPU from becoming flooded by ARP packets in a busy network, you can restrict the number of ARP packets the device will accept each second. When you configure an ARP rate limit, the device accepts up to the maximum number of packets you specify, but drops additional ARP packets received during the one-second interval. When a new one-second interval starts, the counter restarts at zero, so the device again accepts up to the maximum number of ARP packets you specified, but drops additional packets received within the interval.

**NOTE**
If you want to change a previously configured the ARP rate limiting policy, you must remove the previously configured policy using the no rate-limit-arp command before entering the new policy.

The **no** form of the command disables ARP rate limiting.

## Examples

The following example configures the device to accept up to 100 ARP packets each second.

```
device(config)# rate-limit-arp 100
```

# rate-limit input

Configures a port-based rate-limiting policy.

## Syntax

**rate-limit input fixed** *average-rate* [ **burst** *burst-size* ]

**no rate-limit input fixed** *average-rate* [ **burst** *burst-size* ]

**rate-limit input fixed ethe** *stack**/**slot **/**port average-rate*

**no rate-limit input fixed ethe** *stack**/**slot **/**port average-rate*

## Parameters

**fixed**

Configures fixed rate-limiting policy.

*average-rate*

Specifies the maximum number of kilobits per second (kbps).

**burst** *burst-size*

Specifies the burst size in kilobits.

## Modes

Interface configuration mode

LAG configuration mode

## Usage Guidelines

The **no** form of the command removes rate limiting.

## Examples

The following example configures rate limiting on a port.

```
device(config)# interface ethernet 1/1/2
device(config-if-e1000-1/1/2)# rate-limit input fixed 500
```

# rate-limit output

Configures the maximum rate at which outbound traffic is sent on a port priority queue or on a LAG port.

## Syntax

**rate-limit output shaping** *value* [ **priority** *priority-queue* ]

**no rate-limit output shaping** *value* [ **priority** *priority-queue* ]

**rate-limit output shaping ethe** *stack**/**slot**/**port value* [ **priority** *priority-queue* ]

**no rate-limit output shaping ethe** *stack**/**slot**/**port value* [ **priority** *priority-queue* ]

## Parameters

**shaping** *value*

Specifies the rate-shaping limit.

**ethernet** *stackislot/port*

Specifies the Ethernet port.

**priority** *priority-queue*

Specifies a rate-shaping priority. The value can range from 0 to 7.

## Modes

Interface configuration mode

LAG configuration mode

## Usage Guidelines

The **no** form of the command removes the output rate shaping.

## Examples

The following example configures the maximum rate at which outbound traffic is sent on a port priority queue

```
device(config)# interface ethernet 1/2/1
device(config-if-e1000-1/2/1)# rate-limit output shaping 500 priority 7
```

The following example configures the maximum rate at which outbound traffic is sent on a LAG port.

```
device(config)# lag lag1 static
device(config-lag-lag1)# rate-limit output shaping ethe 1/1/15 651
```

# rate-limit-log

Configures the global level BUM suppression logging interval.

## Syntax

**rate-limit-log** [ *minutes* ]

**[no] rate-limit-log** [ *minutes* ]

## Command Default

The default logging interval 5 minutes.

## Parameters

*minutes*

Specifies the interval, in whole mimutes, between Syslog notifications. The value can be any integer from 1 to 10.

## Modes

Global configuration mode

## Usage Guidelines

Use the **no** form of the command to return to the default value (5 minutes).

## Examples

The following example shows how to set the BUM suppression notification Syslog logging interval to 3 minutes.

```
device(config)# rate-limit-log 3
```

## History

| Release version | Command history |
|---|---|
| 8.0.30h | This command was introduced. |

# rconsole

Use the **rconsole command** to establish a remote console session with a stack member.

## Syntax

**rconsole** *stack-unit*

## Command Default

N/A

## Parameters

*stack-unit*

Stack-unit ID of the remote device

## Modes

Privileged EXEC mode.

## Usage Guidelines

You can terminate a session in any of these ways:

- by entering the **exit** command from the User EXEC level
- by entering the **logout** command at any level.

## Examples

To establish an rconsole session, enter the **rconsole** command as shown:

```
device# rconsole 1
```

In the following example, a remote console session is established with stack unit 2.

```
device# rconsole 2
Connecting to unit 2... (Press Ctrl-O X to exit)
rconsole-2@device# show stack
ID   Type        Role      Mac Address      Prio State    Comment
2  S ICX7450-24P standby   0000.00e2.ba40            0    local    Ready
rconsole-2@device# exit
rconsole-2@device> exit
Disconnected.  Returning to local session...
```

## History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |

# rconsole (SPX)

Establishes a remote connection to a control bridge or port extender unit.

## Syntax

**rconsole** { *id* | **controller-bridge** }

## Command Default

By default, a stack member can use the **rconsole** command to connect to the console of the stack's active controller.

## Parameters

*id*

Designates an SPX port extender (PE) unit or CB (core) stack member.

**controller-bridge**

Designates the active controller (master unit) for the CB (core) stack.

## Modes

CB device mode

Stack member device mode

Provisional-PE mode

## Usage Guidelines

The command is available in the same modes as the **show running-config** command.

Use the **rconsole** *id* command on a CB unit to access the local console of the designated PE or CB unit. Use **exit** to terminate the connection.

A stack member or a PE member in an 802.1br CB can access the console of the stack's active controller using the **rconsole controller-bridge** command. Terminate the connection from a stack member to the active controller by pressing **Control+Shift+x**.

Terminate an **rconsole** connection between a CB unit and a PE by entering **Control+o x**.

## Examples

The following example creates a remote connection from the local PE to the active controller of the CB.

```
[PE]local-id@device# rconsole
  controller-bridge   Connect to the active controller bridge
[PE]local-id@device# rconsole controller-bridge
Connecting to control-bridge 3 console... (Press Ctrl-o x to exit)
controller-device>
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# rd

Distinguishes a route for Virtual Routing and Forwarding (VRF) instances.

## Syntax

**rd** {*ASN* **:** *nn* | *IP-address* **:** *nn* }

## Parameters

*ASN:nn*

Configures the RD as autonomous system number followed by a colon (:) and a unique arbitrary number.

*IP-address:nn*

Configures the RD as IP address followed by a colon (:) and a unique arbitrary number.

## Modes

VRF configuration mode

## Usage Guidelines

Each VRF instance is identified by a unique route distinguisher (RD). The RD is prepended to the address being advertised. Because the RD provides overlapping client address space with a unique identifier, the same IP address can be used in different VRFs without conflict. The RD can be an autonomous system number, followed by a colon (:) and a unique arbitrary number as in "10:11". Alternatively, it canbe a local IP address followed by a colon (:) and a unique arbitrary number, as in "1.1.1.1:100".

Once the Route Distinguisher is configured for a VRF it cannot be changed or deleted. To remove the route distinguisher, you must delete the VRF.

## Examples

The following example configures a Route Distinguisher.

```
device(config)# vrf red
sevice(config-vrf-red)# rd 101:101
```

# rear-module

Defines the operating mode of ports on the rear module.

## Syntax

**rear-module** { **stack-40g** | **uplink-100g** | **uplink-40g**}

**no rear-module** { **stack-40g** | **uplink-100g** | **uplink-40g**}

## Command Default

By default, the rear module is used for stacking with two 100-Gbps ports.

## Parameters

**stack-40g**
Configures the rear module for stacking with four 40-Gbps ports.

**uplink-100g**
Configures the rear module for two 100-Gbps uplink ports.

**uplink-40g**
Configures the rear module for two 40-Gbps uplink ports.

## Modes

Global configuration mode

## Usage Guidelines

The command applies to ICX 7650 devices only.

Use the **no** form of the command to return the module to default operation.

100-Gbps and 40-Gbps operation require different optics. If optics do not match the configuration, the port link is down.

Local and remote ports connected through the rear module must operate in the same mode.

The **rear-module** command stacking options are allowed only when no uplink configuration is present on the rear module.The **rear-module** command cannot be used to configure uplink ports once the stack is formed. Refer to the *FastIron Stacking Configuration Guide* for information on enabling stacking or removing configuration.

You are, however, allowed to enter stacking configuration while the rear module operates in uplink mode. The stacking configuration remains offline, and stacking election is blocked. Once you enter the **write memory** command, the system prompts you to save or erase the stacking configuration.

You must execute the **write memory** and **reload** commands for any change to the **rear-module** setting to take effect.

# Examples

The following example configures the rear module to operate in 40-Gbps stacking mode.

```
device# configure terminal
device(config)# rear-module stack-40G
device(config)# exit
device# write memory
device#  reload
```

The following example returns the rear module to 100-Gbps stacking mode (the default).

```
device# configure terminal
device(config)# no rear-module stack-40G
device(config)# exit
device# write memory
device#  reload
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# re-authentication (Flexible authentication)

Periodically re-authenticates clients connected to MAC authentication-enabled interfaces and 802.1X-enabled interfaces.

## Syntax

**re-authentication**

**no re-authentication**

## Command Default

Re-authentication is not enabled.

## Modes

Authentication configuration mode

## Usage Guidelines

The **no** form of this command disables re-authentication.

When periodic reauthentication is enabled, the device reauthenticates the clients every 3,600 seconds by default. The reauthentication interval configured using the **reauth-period** command takes precedence.

## Examples

The following example configures periodic re-authentication using the default interval of 3,600 seconds.

```
device(config)# authentication
device(config-authen)# re-authentication
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 8.0.40a | Reauthentication support was added to MAC authentication-enabled ports. |

# reauth-period

Configure the interval at which clients connected to MAC authentication-enabled ports and 802.1X authentication-enabled ports are periodically reauthenticated.

## Syntax

**reauth-period** *seconds*

**no reauth-period***seconds*

## Command Default

The re-authentication period is 3600 seconds.

## Parameters

*seconds*

Sets the re-authentication period. The range is 1 through 4294967295 seconds.

## Modes

Authentication configuration mode

## Usage Guidelines

While the **re-authentication** command configures periodic re-authentication using the default interval of 3600 seconds, the **reauth-period** command allows you to specify a value in seconds.

The reauthentication interval configured using the **reauth-period** command can be overwritten for each client by the RADIUS server through the Session-Tmeout and Termination-Action attributes.

The **no** form of this command reverts the re-authentication period to the default interval of 3600 seconds.

## Examples

The following example configures periodic re-authentication with an interval of 2,000 seconds.

```
device(config)# authentication
device(config-authen)# re-authentication
device(config-authen)# reauth-period 2000
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 8.0.40a | Reauthentication support was added to MAC authentication-enabled ports. |
| 8.0.30j | Reauthentication support was added to MAC authentication-enabled ports. |

# reauth-time

Configures the number of seconds an authenticated user remains authenticated.

## Syntax

**reauth-time** *seconds*

**no reauth-time** *seconds*

## Command Default

The default is 28,800 seconds.

## Parameters

*seconds*

The number of seconds an authenticated user remains authenticated. The valid values are from 0 through 128,000 seconds. The default is 28,800.

## Modes

Web Authentication configuration mode

## Usage Guidelines

After a successful authentication, a user remains authenticated for a duration of time. At the end of this duration, the host is automatically logged off. The user must be reauthenticated again.

Setting a value of 0 means the user is always authenticated and will never have to reauthenticate, except if an inactive period less than the reauthentication period is configured on the Web Authentication VLAN. If this is the case, the user becomes deauthenticated if there is no activity and the timer for the inactive period expires.

The **no** form of the command sets the value to the default.

## Examples

The following example configures the reauthentication time as 300 seconds.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# reauth-time 300
```

# redistribute

Configures the device to redistribute IPv4 and IPv6 routes from one routing domain to another.

## Syntax

**redistribute** { **ospf** } [ **match** [ **external1** | **external2** | **internal** ] | **metric** *num* | **route-map** *string* ]

**redistribute** { *source-protocol* } [ **metric** *num* | **metric-type** { **type1** | **type2** } | **route-map** *string* ]

**no redistribute** { **ospf** } [ **match** [ **external1** | **external2** | **internal** ] ] [ **metric** *num* ] [ **route-map** *string* ]

**no redistribute** { *source-protocol* } [ **metric** *num* ] [ **metric-type** { **type1** | **type2** } ] [ **route-map** *string* ]

## Command Default

The device does not redistribute routing information.

## Parameters

**match**

Specifies the type of route.

**external1**

Specifies OSPF Type 1 external routes.

**external2**

Specifies OSPF Type 2 external routes.

**internal**

Specifies OSPF internal routes.

*source-protocol*

Specifies the source protocol from which routes are being redistributed. It can be one of the following keywords: **bgp**, **connected**, **ospf**, **rip**, or **static**.

**metric** *num*

Specifies a metric for redistributed routes. Range is from 0 through 65535. No value is assigned by default.

**route-map** *string*

Specifies a route map to be consulted before a route is added to the routing table.

**metric-type**

Specifies the external link type associated with the default route advertised into the OSPF routing domain.

**type1**

Specifies a type 1 external route.

**type2**

Specifies a type 2 external route.

**level-1**

Specifies level-1 routes.

**level-1-2**

Specifies both level-1 and level-2 routes.

**level-2**

Specifies level-2 routes.

# Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

OSPF router configuration mode

OSPFv3 router configuration mode

OSPF router VRF configuration mode

OSPFv3 router VRF configuration mode

# Usage Guidelines

Routes can be filtered by means of an associated route map before they are distributed.

The **metric-type** { **type1** | **type2** } option is only available in OSPFv3 router configuration mode and OSPFv3 router VRF configuration mode.

[ **match metricmetric-type**

**NOTE**
The **default-metric** command does not apply to the redistribution of directly connected routes. Use a route map to change the default metric for directly connected routes.

The **no** form of the command restores the defaults.

# Examples

The following example redistributes OSPF external type 1 routes.

```
device# configure terminal
device(config)# router bgp
device(config-bgp)# redistribute ospf match external1
```

The following example redistributes OSPF routes with a metric of 200.

```
device# configure terminal
device(config)# router bgp
device(config-bgp)# redistribute ospf metric 200
```

The following example redistributes OSPFv3 external type 2 routes in VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp)# address-family ipv6 unicast vrf red
device(config-bgp-ipv6u-vrf)# redistribute ospf match external2
```

The following example redistributes static routes into BGP4+ and specifies a metric of 200.

```
device# configure terminal
device(config)# router bgp
device(config-bgp)# address-family ipv6 unicast
device(config-bgp-ipv6u)# redistribute static metric 200
```

The following example redistributes RIP routes and specifies that route-map "rm2" be consulted in BGP address-family IPv6 unicast configuration mode.

```
device# configure terminal
device(config)# router bgp
device(config-bgp)# address-family ipv6 unicast
device(config-bgp-ipv6u)# redistribute rip route-map rm2
```

The following example redistributes BGP routes and specifies that route-map "rm7" be consulted in OSPF router configuration mode.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# redistribute bgp route-map rm7
```

The following example redistributes OSPF routes and specifies a type1 external route in OSPFv3 VRF configuration mode.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# redistribute ospf metric-type type1
```

# redistribute (BGP)

Configures the device to redistribute RIP routes, directly connected routes, or static routes into BGP4 and BGP4+.

## Syntax

**redistribute** { **connected** | **rip** | **static** } [ **metric** *num* ] [ **route-map** *string* ]

**no redistribute** { **connected** | **rip** | **static** } [ **metric** *num* ] [ **route-map** *string* ]

## Command Default

The device does not redistribute routing information between BGP4 or BGP4+ and the IP interior gateway protocol OSPF.

## Parameters

**connected**
Redistributes connected routes.

**rip**
Redistributes Routing Information Protocol (RIP) routes.

**static**
Redistributes static routes.

**metric**
Metric for redistributed routes.

**num**
Specifies a metric number. The range is from 0 through 4294967297. No value is assigned by default.

**route-map**
Specifies that a route map be consulted before a route is added to the routing table.

*string*
Specifies a route map to be consulted before a route is added to the routing table.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

Use the **no** form of the command to restore the defaults. When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

Use this command to configure the device to redistribute RIP, directly connected routes, or static routes into BGP4 or BGP4+. The routes can be filtered by means of an associated route map before they are distributed.

**NOTE**
The **default-metric** command does not apply to the redistribution of directly connected routes into BGP4 or BGP4+. Use a route map to change the default metric for directly connected routes.

# Examples

This example redistributes static routes into BGP4 and specifies a metric of 200.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# redistribute static metric 200
```

This example redistributes static routes into BGP4+ and specifies that route-map "rm5" be consulted.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# redistribute route-map rm5
```

This example redistributes directly connected routes into BGP4 in VRF instance "red".

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv4 unicast vrf red
device(config-bgp-ipv4u-vrf)# redistribute connected
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | Support was added for the BGP address-family IPv6 unicast VRF configuration mode. |

# redistribute (RIP)

Configures the device to redistribute connected routes, learned static routes, OSPF routes, or BGP4 routes through RIP. The RIP router can then advertise these routes to RIP neighbors.

## Syntax

**redistribute** { **connected** | **bgp** | **ospf** | **static** [ **metric** *value* | **route-map** *name* ] }

**no redistribute** { **connected** | **bgp** | **ospf** | **static** [ **metric** *value* | **route-map** *name* ] }

## Command Default

By default, redistribution of other routes is disabled. Once redistribution of a particular type of route is enabled, the default action is to permit redistribution, even with redistribution filters applied to the virtual routing interface.

## Parameters

**connected**
Redistributes connected routes.

**bgp**
Redistributes BGP routes.

**ospf**
Redistributes OSPF routes.

**static**
Redistributes IP static routes.

**metric**
Sets a RIP route metric to the value specified.

*value*
Specifies the RIP route metric as a value from 1 through 15.

**route-map**
Applies the specified route map to routes designated for redistribution.

*name*
Specifies the route-map to be applied.

## Modes

RIP router configuration mode.

## Usage Guidelines

The **no** form of the command removes redistribution actions specified in the command.

To control redistribution tightly, apply a filter to deny all routes and give it the highest ID. Then apply filters to allow specific routes.

RIP redistribution filters apply to all interfaces. Use route maps to define where to deny or permit redistribution. Refer to the route-map command for information on configuring route maps for RIP.

# Examples

The following example redistributes connected routes and adds 10 to the metric for each route.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# redistribute connected metric 10
```

The following example discontinues redistribution and the added metric applied in the previous example.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# no redistribute connected metric 10
```

The following example redistributes all connected route types based on the specifics of the route map named routemap1.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# redistribute connected route-map routemap1
```

# redistribute (RIPng)

Configures RIPng to advertise routes from the specified protocol or connections.

## Syntax

**redistribute** { **bgp** | **connected** | **ospf** | **static** [ **metric** *value* ] }

**no redistribute** { **bgp** | **connected** | **ospf** | **static** [ **metric** *value* ] }

## Command Default

By default, routes from these protocols are not shared between RIPng neighbors.

## Parameters

**connected**
Redistributes directly connected IPv6 network routes.

**bgp**
Redistributes BGP4+ routes.

**ospf**
Redistributes OSPFv3 routes.

**static**
Redistributes IPv6 static routes.

**metric**
Sets a RIPng route metric to the value specified. When no metric is set, the default metric of one is used.

*value*
Specifies RIPng route metric as a value from1 through 15.

## Modes

RIPng router configuration mode.

## Usage Guidelines

The **no** form of the command removes redistribution actions specified in the command.

## Examples

The following example configures the RIPng router to redistribute OSPF routes.

```
device# configure terminal
device(config)# ipv6 router rip
device(config-ripng-router)# redistribute ospf
```

# regenerate-seq-num

Changes the sequence numbers of the rules within a specified access control list (ACL) to provide flexibility in inserting new rules between existing rules.

## Syntax

**regenerate-seq-num** [ *start-sequence-value* [ *increment-value* ] ]

## Parameters

*start-sequence-value*

Specifies the sequence number assigned to the first rule. Values range from 1 through 65000. The default is 10.

*increment-value*

Specifies the increment between the regenerated sequence numbers. Values range from 1 through 100. The default is 10.

## Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

## Usage Guidelines

This command is effective for all IPv4 ACL types: named or numbered; standard or extended.

This command is effective for IPv6 ACLs.

After resequencing, you do not need to rebind the ACL.

From FastIron 08.0.61, sequence regeneration settings (first sequence number and sequence interval number) are persistent, even following reload of the active unit.

## Examples

The following example regenerates sequence numbers for a standard numbered IPv4 ACL.

```
device# configure terminal
device(config)# ip access-list standard 18
device(config-std-nacl)# regenerate-seq-number
```

The following example regenerates sequence numbers for an extended named IPv4 ACL, specifying that the first rule be numbered 100, with an increment of 15 between each sequence number.

```
device# configure terminal
device(config)# ip access-list extended extACLtest_01
device(config-ext-nacl)# regenerate-seq-number 100 15
```

The following example regenerates sequence numbers for an IPv6 ACL.

```
device# configure terminal
device(config)# ipv6 access-list ACL6_01
device(config-ipv6-acces-list ACL6_01)# regenerate-seq-number
```

The following example demonstrates that non-default sequence regeneration is persistent following reload of the active unit.

```
device# configure terminal
device(config)# ip access-list extended testACL
device(config-ext-nacl)# permit ip host 1.1.1.111 host 2.2.2.111
device(config-ext-nacl)# permit ospf any any
device(config-ext-nacl)# permit pim any any

device(config-ext-nacl)# show ip access-lists testACL
Extended IP access list testACL: 3 entries
10: permit ip host 1.1.1.111 host 2.2.2.111
20: permit ospf any any
30: permit pim any any

device(config-ext-nacl)# regenerate-seq-number 100 100

device(config-ext-nacl)# show ip access-lists testACL
Extended IP access list testACL: 3 entries
100: permit ip host 1.1.1.111 host 2.2.2.111
200: permit ospf any any
300: permit pim any any

device(config-ext-nacl)# sequence 150 deny ip 20.20.20.96 0.0.0.15 any

device(config-ext-nacl)# show ip access-lists testACL
Extended IP access list testACL: 4 entries
100: permit ip host 1.1.1.111 host 2.2.2.111
150: deny ip 20.20.20.96 0.0.0.15 any
200: permit ospf any any
300 permit pim any any

<Reload of active unit>

device(config-ext-nacl)# show ip access-lists testACL
Extended IP access list testACL: 4 entries
100: permit ip host 1.1.1.111 host 2.2.2.111
150: deny ip 20.20.20.96 0.0.0.15 any
200: permit ospf any any
300 permit pim any any
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.61 | The command has been modified so that a non-default run of the command is persistent following device reload. |

# register-probe-time

Configures the time the PIM router waits for a register-stop from a rendezvous point (RP) before it generates another NULL register to the PIM RP

## Syntax

**register-probe-time** *seconds*

**no register-probe-time** *seconds*

## Command Default

The wait time is10 seconds.

## Parameters

*seconds*

Specifies the time, in seconds, between queries. The range is 10 through 50 seconds. The default is 10 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the wait time to 10 seconds.

The register-probe time configuration applies only to the first-hop PIM router.

**NOTE**
When a PIM first-hop router has successfully registered with a PIM RP, the PIM first-hop router will not default back to the data registration. All subsequent registers will be in the form of the NULL registration.

## Examples

This example configures the register-probe time to 20 seconds.

```
Device(config)#router pim
Device(config-pim-router)#register-probe-time 20
```

# register-suppress-time

Configures the interval at which the PIM router triggers the NULL register message.

## Syntax

**register-suppress-time** *seconds*

**no register-suppress-time** *seconds*

## Command Default

The interval at which PIM router triggers the NULL register message is 60 seconds.

## Parameters

*seconds*

Specifies the interval, in seconds, between queries. The range is 60 through 120 seconds. The default is 60 seconds.

## Modes

PIM router configuration mode

## Usage Guidelines

The **no** form of this command restores the register-suppress interval to 60 seconds.

The register-suppress interval configuration applies only to the first-hop PIM router.

## Examples

The following example configures the interval at which PIM router triggers the NULL register message to 90 seconds.

```
Device(config)#router pim
Device(config-pim-router)#register-suppress-time 90
```

# relative-utilization

Configures uplink utilization lists that display the percentage of a given uplink port's bandwidth that is used by a specific list of downlink ports.

## Syntax

**relative-utilization** *number* **uplink ethernet** *stack-id/slot/port* [ **to** *stack-id/slot/port* | [ **ethernet** *stack-id/slot/port* **to** *stack-id/slot/port* | **ethernet** *stack-id/slot/port* ] ... ] **downlink ethernet** *stack-id/slot/port* [ **to** *stack-id/slot/port* | [ **ethernet** *stack-id/slot/port* **to** *stack-id/slot/port* | **ethernet** *stack-id/slot/port* ] ... ]

**no relative-utilization** *number* **uplink ethernet** *stack-id/slot/port* [ **to** *stack-id/slot/port* | [ **ethernet** *stack-id/slot/port* **to** *stack-id/slot/port* | **ethernet** *stack-id/slot/port* ] ... ] **downlink ethernet** *stack-id/slot/port* [ **to** *stack-id/slot/port* | [ **ethernet** *stack-id/slot/port* **to** *stack-id/slot/port* | **ethernet** *stack-id/slot/port* ] ... ]

## Command Default

Relative utilization is not configured.

## Parameters

*number*
Specifies the list number. The value can range from 1 to 4. You can specify upto four lists.

**uplink ethernet** *stack-id/slot/port*
Specifies the uplink Ethernet port.

**to** *stack-id/slot/port*
Specifies a range of Ethernet ports.

**downlink ethernet** *stack-id/slot/port*
Specifies the downlink Ethernet port.

## Modes

Global configuration mode

## Usage Guidelines

Each uplink utilization list consists of the following:

- Utilization list number (1, 2, 3, or 4).
- One or more uplink ports.
- One or more downlink ports.

Each list displays the uplink port and the percentage of that port's bandwidth that was utilized by the downlink ports over the most recent 30-second interval.

You can configure up to four bandwidth utilization lists.

You can specify a list or range of ports as uplink or downlink ports.

The **no** form of the command removes the uplink utilization list.

# Examples

The following example configures an uplink utilization list.

```
device(config)# relative-utilization 1 uplink ethernet 1/1/1 downlink ethernet 1/2/2 to 1/3/2
```

# reload

Reloads a stand-alone device, a stack, or specified stack units other than the active controller.

## Syntax

**reload** [ **after** *duration* | **at** *time date* [ **primary** | **secondary** ] ]

**reload cancel**

**reload** [ **unit-id** *unit-list* ]

## Parameters

**after** *duration*

Schedules reload after the specified duration, entered in the format *dd:hh:mm*, where *dd* is the number of days; *hh* represents the number of hours, from 00 to 23; and *mm* represents minutes, from 00 to 59.

**at** *time date*

Schedules reload for a specific time and date. Time is entered in this format: *hh:mm:ss*, where *hh* represents hours, from 00 to 24; *mm* represents minutes, from 00 to 59; and *ss* represents seconds, from 00 to 59. The date is entered in this format: *mm-dd-yy*, where *mm* is the month (for example, 01 for January); *dd* is the day in the month (for example, 09); and *yy* is the year (for example, 17).

**cancel**

Cancels the scheduled stack reload.

**primary**

Reloads from primary image flash.

**secondary**

Reloads from secondary image flash.

**unit-id** *unit-list*

Specifies stack units to reload. When the **unit-id** is not present, the stand-alone device or the stack on which the **reload** command is issued is reloaded. The *unit-list* may contain a single ID (2), a series of IDs (2,3), a range of IDs (4-6), or a combination ( 2,3,4-6,8). Do not use spaces between entries.

## Modes

Privileged EXEC mode

## Usage Guidelines

Stack units can be reloaded only if they are not the active controller.

The active controller automatically reloads on stack failover to the standby controller. If you need to reload the active controller manually, use the **stack switch-over** command. When switchover occurs, you will be able to load the former active controller with the **reload** *unit-id* command.

# Examples

When the **reload** command is entered on the active controller without the **unit-id** parameter as shown in the following example, the entire stack reloads. When the **reload** command is entered on a stand-alone device without a unit ID, the device reloads.

```
device# reload
```

# History

| Release version | Command history |
|---|---|
| FastIron release 08.0.00a | This command was introduced. |

# remark

Adds a comment to describe entries in an IPv4 or IPv6 ACL.

## Syntax

**remark** *comment-text*

**no remark** *comment-text*

## Command Default

No comments are added to describe entries in an IPv4 or IPv6 ACL.

## Parameters

*comment-text*

Specifies the comment for the ACL entry, up to 256 alphanumeric characters.

## Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

## Usage Guidelines

You can add a comment by entering the **remark** command immediately preceding an ACL entry. The comment appears in the output of show commands that display ACL information.

The **no** form of the command deletes the comment text added for an ACL entry.

## Examples

The following example configures remarks for an IPv4 ACL.

```
device(config)# ip access-list extended TCP/UDP
device(config-ext-nacl)# remark The following line permits TCP packets
device(config-ext-nacl)# permit tcp 192.168.4.40/24 2.2.2.2/24
device(config-ext-nacl)# remark The following permits UDP packets
device(config-ext-nacl)# permit udp 192.168.2.52/24 2.2.2.2/24
device(config-ext-nacl)# deny ip any any
```

The following example configures remarks for an IPv6 ACL.

```
device(config)# ipv6 access-list rtr
device(config-ipv6-access-list rtr)# remark This entry permits ipv6 packets from 2001:DB8::2 to any
destination
device(config-ipv6-access-list rtr)# permit ipv6 host 2001:DB8::2 any
device(config-ipv6-access-list rtr)# remark This entry denies udp packets from any source to any
destination
device(config-ipv6-access-list rtr)# deny udp any any
device(config-ipv6-access-list rtr)# remark This entry denies IPv6 packets from any source to any
destination
device(config-ipv6-access-list rtr)# deny ipv6 any
```

The following example shows the comment text for the ACL named "rtr" in a show running-config display.

```
device# show running-config
ipv6 access-list rtr
remark This entry permits ipv6 packets from 2001:DB8::2 to any destination permit ipv6 host 2001:DB8::2
any
remark This entry denies udp packets from any source to any destination deny udp any any
remark This entry denies IPv6 packets from any source to any destination deny ipv6 any any
```

The following example shows how to delete a comment from an IPv6 ACL entry.

```
device(config)# ipv6 access-list rtr
device(config-ipv6-access-list rtr)# no remark This entry permits ipv6 packets from 2001:DB8::2 to any
destination
```

# remote-identifier

Configures a remote identifier for an Internet Key Exchange version 2 (IKEv2) profile.

## Syntax

**remote-identifier** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

**no remote-identifier** { **address** { *ip-address* | *ipv6-address* } | **dn** *dn-name* | **email** *email-address* | **fqdn** *fqdn-name* | **key-id** *key-id* }

## Command Default

A remote identifier is not configured for an IKEv2 profile.

## Parameters

**address** *ip-address*
Specifies an IPv4 address as the remote identifier.

**address** *ipv6-address*
Specifies an IPv6 address as the remote identifier.

**dn** *dn-name*
Specifies a Distinguished Name (DN) as the remote identifier.

**email** *email-address*
Specifies an email address as the remote identifier.

**fqdn** *fqdn-name*
Specifies a fully qualified domain name (FQDN) as the remote identifier.

**key-id** *key-id*
Specifies a key ID as the remote identifier.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

The **no** form of the command removes the remote identifier configuration.

## Examples

The following example shows how to configure IPv4 address 10.2.2.1 as the remote identifier for an IKEv2 profile named prof_mktg.

```
device(config)# ikev2 profile prof-mktg
device(config-ike-profile-prof-mktg)# remote-identifier address 10.2.2.1
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support was added for IPv6. |

# remote-loopback

Starts or stops the remote loopback procedure on a remote device.

## Syntax

**remote-loopback ethernet** *stackid/slot/port* { **start** | **stop** }

## Command Default

Remote loopback is not initiated on a remote device.

## Parameters

**ethernet** *stackid/slot/port*
Specifies the Ethernet interface on which loopback is to be enabled.

**start**
Starts the remote loopback procedure on a remote device.

**stop**
Stops the remote loopback procedure on a remote device.

## Modes

EFM-OAM protocol configuration mode

## Usage Guidelines

The **remote-loopback ethernet** *stackid/slot/port* { **start** | **stop** } command is valid only on the Data Terminal Equipment (DTE) operating in the active mode.

When the remote loopback mode is enabled, all the non-OAMPDUs are looped back at the remote end.

A port ceases to be in the remote loopback mode if any event triggers a change in the port status (up or down).

If EEE is enabled globally, port ceases to be in the remote loopback mode.

Ethernet loopback and EFM-OAM remote loopback cannot be configured on the same interface.

**NOTE**
Ruckus recommends you ensure that any higher layer protocol running over the local and remote loopback ports does not block the interfaces in the VLAN on which loopback traffic testing is being performed.

## Examples

The following example initiates the remote loopback procedure on a remote DTE.

```
device(config)# link-oam
device(config-link-oam)# remote-loopback ethernet 3/1/1 start
```

The following example stops the remote loopback procedure on a remote DTE.

```
device(config)# link-oam
device(config-link-oam)# remote-loopback ethernet 3/1/1 stop
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# remove-tagged-ports

Removes all tagged member ports from a VLAN or from multiple VLANs.

## Syntax

**remove-tagged-ports**

## Modes

VLAN configuration mode

Multiple VLAN configuration mode

## Examples

The following example removes all tagged member ports from VLAN 2.

```
device(config)# show run vlan 2
vlan 2 by port
tagged ethernet 1/1/1 to 1/1/3
untagged ethernet 1/1/4 to 1/1/16
!
!
device(config)# vlan 2
device(config-vlan-2)# remove-tagged-ports
device(config-vlan-2)# show run vlan 2
vlan 2 by port
untagged ethernet 1/1/4 to 1/1/6
!
!
```

The following example removes all tagged member ports from a range of VLANs.

```
device(config)# show run vlan
vlan 1 name DEFAULT-VLAN by port
!
vlan 2 by port
tagged ethernet 1/1/1 to 1/1/5
untagged ethernet 1/1/6 to 1/1/7
!
vlan 3 by port
tagged ethernet 1/1/1 to 1/1/5
untagged ethernet 1/1/8 to 1/1/9
!
!
device(config)# vlan 2 3
device(config-mvlan-2-3)# remove-tagged-ports
device(config-mvlan-2-3)# show run vlan
vlan 2 by port
untagged ethernet 1/1/6 to 1/1/7
!
vlan 3 by port
untagged ethernet 1/1/8 to 1/1/9
!
!
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |

# remove-untagged-ports

Removes all untagged member ports from a VLAN or from multiple VLANs.

## Syntax

**remove-untagged-ports**

## Modes

VLAN configuration mode

Multiple VLAN configuration mode

## Examples

The following example removes all untagged member ports from VLAN 2.

```
device(config)# show run vlan 2
vlan 2 by port
tagged ethernet 1/1/1 to 1/1/3
untagged ethernet 1/1/4 to 1/1/16
!
!
device(config)# vlan 2
device(config-vlan-2)# remove-untagged-ports
device(config-vlan-2)# show run vlan 2
vlan 2 by port
tagged ethernet 1/1/1 to 1/1/3
!
!
```

The following example removes all untagged member ports from a range of VLANs.

```
device(config)# show run vlan
vlan 2 by port
tagged ethernet 1/1/1 to 1/1/3
untagged ethernet 1/1/4 to 1/1/16
!
!
vlan 3 by port
tagged ethernet 1/1/1 to 1/1/3
untagged ethernet 1/1/4 to 1/1/16
!
!
device(config)# vlan 2 3
device(config-mvlan-2-3)# remove-untagged-ports
device(config-vlan-2-3)# show run vlan
vlan 2 by port
tagged ethernet 1/1/1 to 1/1/3
!
!
vlan 3 by port
tagged ethernet 1/1/1 to 1/1/3
!
!
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |

# remove-vlan(VLAN group)

Removes individual VLANs or a range of VLANs from a VLAN group.

## Syntax

**remove-vlan** *vlan-id* [ **to** *vlan-id* ]

## Parameters

*vlan-id*

Specifies the VLAN number to remove from a VLAN group.

to *vlan-id*

Specifies the range of VLAN numbers to remove from a VLAN group.

## Modes

VLAN group configuration mode

## Usage Guidelines

Use the **vlan-group** command to create a range of VLANs. To remove one or more VLANs from a VLAN group, use the **remove-vlan** command.

## Examples

The following example removes the specified VLANs from vlan-group 1.

```
device(config)# vlan-group 1 vlan 10 to 15
device(config-vlan-group-1)# remove-vlan 10
device(config-vlan-group-1)# remove-vlan 11 to 12
device(config-vlan-group-1)# show vlan-group
vlan group 1 vlan 13 to 15
!
!
```

## History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.50 | This command is no longer supported in Interface Configuration mode. |

# replay-protection

Used with extended sequence numbering in IPsec to prevent replay attacks by assigning each encrypted packet an increasing sequence number that is tracked at the IPsec endpoint.

## Syntax

**replay-protection**

**no replay-protection**

## Command Default

Anti-replay protection is disabled by default.

## Modes

IPsec profile configuration sub-mode

## Usage Guidelines

The **replay-protection** command must be used in conjunction with extended sequence numbering (ESN), which is configured with the **esn-enable** command in the IPsec proposal.

The **no** form of the command disables anti-replay protection.

## Examples

The following example configures IPsec anti-replay protection as part of the IPsec profile ipsecprof1.

```
device# configure terminal
device(config)# ipsec profile ipsecprof1
device(config-ipsec-profile-ipsecprof1)# replay-protection
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# reserved-vlan-map

Assigns a different VLAN ID to the reserved VLAN.

## Syntax

**reserved-vlan-map vlan** *vlan-id* **new-vlan** *vlan-id*

**no reserved-vlan-map vlan** *vlan-id* **new-vlan** *vlan-id*

## Command Default

The reserved VLAN ID are 4091 and 4092.

## Parameters

**vlan** *vlan-id*

Specifies the default reserved VLAN ID.

**new-vlan** *vlan-id*

Specifies the new VLAN ID that you want to assign to the reserved VLAN.

## Modes

Global configuration mode

## Usage Guidelines

For *vlan-id*, enter a valid VLAN ID that is not already in use. Valid VLAN IDs are numbers from 1 through 4090, 4093, and 4095. VLAN ID 4094 is reserved for use by Single STP.

**NOTE**

You must save the configuration (**write memory**) and reload the software to place the change into effect.

The **no** form of the command resets the values back to the default reserved VLAN IDs.

## Examples

The following example shows how to assign a new VLAN ID to the reserved VLAN IDs.

```
device(config)# reserved-vlan-map vlan 4091 new-vlan 10
Reload required. Please write memory and then reload or power cycle.
device(config)# write memory
device(config)# exit
device# reload
```

# responder-only

Configures responder-only mode for an IKEv2 profile.

## Syntax

**responder-only**

**no responder-only**

## Command Default

The responder-only mode is disabled.

## Modes

IKEv2 profile configuration mode

## Usage Guidelines

By default responder-only mode is disabled and the device behaves as both initiator and responder so that IKEv2 negotiations start when the IKEv2 peer is reachable.

In responder-only mode, the device is passive and does not initiate negotiation or re-keying to establish an IKEv2 security association (SA).

The **no** form of the command disables responder-only mode.

## Examples

The following example enables responder-only mode for an IKEv2 profile named ikev2_profile1.

```
device# configure terminal
device(config)# ikev2 profile ikev2_profile1
device(config-ike-profile-ikev2_profile1)# responder-only
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# restart-ports

Configures a VSRP-configured device to shut down its ports when a failover occurs and restart after a period of time.

## Syntax

**restart-ports** [ *seconds* ]

**no restart-ports** *seconds*

## Command Default

The default is 1 second.

## Parameters

*seconds*

Specifies the time the VSRP master shuts down its port before it restarts. The range is from 1 through 120 seconds.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

The VSRP fast start feature can be enabled on a VSRP-configured Ruckus device, either on the VLAN to which the VRID of the VSRP-configured device belongs (globally) or on a port that belongs to the VRID. This command shuts down all the ports that belong to the VLAN when a failover occurs. All the ports will have the specified VRID.

The **no** form of the command resets the time to the default.

## Examples

The following example configures the ports to restart in 5 seconds.

```
device(config)# vlan 100
device(config-vlan-100)# vsrp vrid 1
device(config-vlan-100-vrid-1)# restart-ports 5
```

# restart-vsrp-port

Configures a single port on a VSRP-configured device to shut down when a failover occurs and restart after a period of time.

## Syntax

**restart-vsrp-port** *seconds*

**no restart-vsrp-port** *seconds*

## Command Default

The default is 1 second.

## Parameters

*seconds*

Configures the VSRP master to shut down its port for the specified number of seconds before it restarts. The range is from 1 through 120 seconds.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command resets the time to the default.

## Examples

The following example configures the VSRP port to restart in 5 seconds.

```
device(config)# interface ethernet 1/1/1
device(config-if-e-10000)# restart-vsrp-port 5
```

# restricted-vlan

Configures a specific VLAN as the restricted VLAN for all ports on the device to place the client port when the authentication fails.

## Syntax

**restricted-vlan** *vlan-id*

**no restricted-vlan** *vlan-id*

## Command Default

The restricted VLAN is not configured.

## Parameters

*vlan-id*

Specifies the identification number of the restricted VLAN.

## Modes

Authentication configuration mode

## Usage Guidelines

When an authentication fails, the port can be moved into a configured restricted VLAN instead of blocking the client completely. The port is moved to the configured restricted VLAN only if the authentication failure action is set to place the port in a restricted VLAN using the **auth-fail-action** command at the global level or using the **authentication fail-action** command at the interface level. Else, when the authentication fails, the client's MAC address is blocked in the hardware (default action).

The **no** form of the command disables the restricted VLAN.

## Examples

The following example creates a restricted VLAN with VLAN 4.

```
device(config)# authentication
device(config-authen)# restricted-vlan 4
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# reverse-manifest-enable

Enables the system backup to USB operation to be carried on the system.

## Syntax

**reverse-manifest-enable**

**no reverse-manifest-enable**

## Modes

Global configuration mode

## Usage Guidelines

To initiate system backup to USB, you must plug in the USB drive when the system is up and running, and press and hold the USB mode button for 10 seconds.

The **no** form of the command disables system backup operation.

## Examples

The following example enables system backup to USB.

```
device(config)# reverse-manifest-enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# reverse-path-check

Enables uRPF for all Layer 3 routes.

## Syntax

**reverse-path-check**

**no reverse-path-check**

## Command Default

Reverse path check is not enabled on the device.

## Modes

Global configuration mode

## Usage Guidelines

On ICX devices, this command enables the uRPF command line interface and hardware settings.

You must reload the device for the reverse path check setting changes to take effect. Enabling reverse path check on ICX devices reduces the following system-max values by half:

- ip-route
- ip6-route
- ip-route-default-vrf
- ip6-route-default-vrf
- ip-route-vrf
- ip6-route-vrf

You should configure these values after reloading. You should adjust or remove the max-route configuration in VRFs before reload.

The **no** form of the command disables the reverse path check functionality.

**NOTE**
Disabling reverse path check doubles the system-max values on ICX devices.

## Examples

The following example enables unicast Reverse Path Forwarding globally.

```
device(config)# reverse-path-check
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |
| 8.0.40 | Removed reference to ICX 6610 devices. |
| 8.0.50 | Removed reference to ICX 7750 devices, because uRPF is supported on all ICX devices. |

# revocation-check (PKI)

Specifies the method to be used for certificate revocation checks.

## Syntax

**revocation-check** { **crl** | **ocsp** | **none** }

**no revocation-chec k** { **crl** | **ocsp** | **none** }

## Command Default

**revocation-check none**

## Parameters

**crl**

Sets the revocation check method to Certificate Revocation List (CRL).

**ocsp**

Sets the revocation check method to Online Certificate Status Protocol (OCSP).

**none**

Designates that no revocation check is to be done. This is the default.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

## Examples

The following example sets the revocation check method for trustpoint abcd to Online Certificate Status Protocol (OCSP).

```
device# configure terminal
device(config)# pki trustpoint abcd
device(config-pki-trustpoint-abcd)# ocsp http post
device(config-pki-trustpoint-abcd)# revocation-check ocsp
device(config-pki-trustpoint-abcd)# ocsp-url http://10.21.40.39:2560
device(config-pki-trustpoint-abcd)# fingerprint 3C:EA:EC:E6:F1:DD:3B:86:65:DE:58:F4:A2:75:D8:63:6D:
23:68:40
device(config-pki-trustpoint-abcd)# exit
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# rfc1583-compatibility (OSPF)

Configures compatibility with RFC 1583.

## Syntax

**rfc1583-compatibility**

**no rfc1583-compatibility**

## Command Default

OSPF is compatible with RFC 1583 (OSPFv2).

## Modes

OSPF router configuration mode

OSPF router VRF configuration mode

## Usage Guidelines

OSPF is compatible with RFC 1583 (OSPFv2) and maintains a single best route to an autonomous system (AS) boundary router in the OSPF routing table. Disabling this compatibility causes the OSPF routing table to maintain multiple intra-AS paths, which helps prevent routing loops.

Enter **no rfc1583-compatibility** to disable compatibility with RFC 1583.

## Examples

The following example disables compatibility with RFC 1583.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# no rfc1583-compatibility
```

# ring-interfaces

Configures the primary and secondary interfaces for the ring to control outward traffic flow.

## Syntax

**ring-interfaces** { **ethernet** *unit/slot/port* | **lag** *lag-id* } { **ethernet** *unit/slot/port* | **lag** *lag-id* }

**no ring-interfaces** { **ethernet** *unit/slot/port* | **lag** *lag-id* } { **ethernet** *unit/slot/port* | **lag** *lag-id* }

## Command Default

The primary and secondary interfaces are not configured.

## Parameters

**ethernet** *unit/slot/port*
: Configures the primary and secondary interfaces.

**lag** *lag-id*
: Specifies the LAG virtual interface.

## Modes

MRP configuration mode

## Usage Guidelines

On the master node, the primary interface is the one that originates Ring Health Packets (RHPs). Ring control traffic and Layer 2 data traffic will flow in the outward direction from this interface by default. On member nodes, the direction of traffic flow depends on the traffic direction selected by the master node. Therefore, on a member node, the order in which you enter the interfaces does not matter.

The **no** form of the command clears the primary and secondary interfaces.

## Examples

The following example shows how to configure the primary and secondary interfaces on a ring.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-2)# ring-interface ethernet 1/1/1 ethernet 1/1/2
```

The following example shows how to configure the LAG virtual interfaces on a ring.

```
device(config)# vlan 2
device(config-vlan-2)# metro-ring 1
device(config-vlan-2-mrp-2)# ring-interface lag 1 lag 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# rmon alarm

Configures an Remote Monitoring (RMON) alarm.

## Syntax

**rmon alarm** *alarm-num mib-object sample-interval* { **absolute** | **delta** } **falling-threshold** *falling-threshold-value event* **rising-threshold** *rising-threshold-value event* **owner** *alarm-owner*

**no rmon alarm** *alarm-num mib-object sample-interval* { **absolute** | **delta** } **falling-threshold** *falling-threshold-value event* **rising-threshold** *rising-threshold-value event* **owner** *alarm-owner*

## Command Default

An RMON alarm is not configured.

## Parameters

*alarm-num*
Specifies the alarm number. The value can range from 1 to 65535.

*mib-object*
Specifies the MIB object to monitor.

*sample-interval*
Specifies the sample interval.

**absolute**
Configures testing each sample directly.

**delta**
Configures testing the delta between the samples.

**falling-threshold**
Configures the falling threshold.

*falling-threshold-value*
Specifies the faliing threshold value. The value can range from 0 to 2147483647.

*event*
Specifies the action (event) to take to fire when the falling threshold crosses the configured value. The value can range from 1 through 65535.

**rising-threshold**
Configures the rising threshold.

*rising-threshold-value*
Specifies the threshold value. The value can range from 0 to 2147483647.

*event*
Specifies the event to fire when the rising threshold crosses the configured value. The value can range from 1 through 65535.

**owner** *alarm-owner*
Specifies the alarm owner.

# Modes

Global configuration mode

# Usage Guidelines

An Alarm is designed to monitor configured thresholds for any SNMP integer, time tick, gauge, or counter MIB object. Using the CLI, you can define what MIB objects are monitored, the type of thresholds that are monitored (falling, rising, or both), the value of those thresholds, and the sample type (absolute or delta).

An alarm event is reported each time that a threshold is exceeded. The alarm entry also indicates the action (event) to be taken if the threshold be exceeded.

You can configure both the falling threshold and the rising threshold and in any order.

The **no** form of the command removes the configured RMON alarm.

# Examples

The following example configures an alarm.

```
device(config)# rmon alarm 1 ifInOctets.6 10 delta rising-threshold 100 1 falling-threshold 50 1 owner
nyc02
```

# rmon event

Defines the action to be taken when an alarm is reported and collects and stores reported events for retrieval by an Remote Monitoring (RMON) application.

## Syntax

**rmon event** *event-entry* **description** *event-description* { { **execute** | **log-and-execute** | **log-trap-and-execute** | **trap-and-execute** } [ **argument** *string* ] | **log** | **trap** | **log-and-trap**} **owner** *event-owner*

**no rmon event** *event-entry* **description** *event-description* { { **execute** | **log-and-execute** | **log-trap-and-execute** | **trap-and-execute** } [ **argument** *string* ] | **log** | **trap** | **log-and-trap**} **owner** *event-owner*

## Command Default

An RMON event is not configured.

## Parameters

*event-entry*
Specifies the event number.

**description** *event-description*
Configures the event description.

**execute**
Executes a batch command when the event fires.

**log-and-execute**
Generates an RMON log and execute batch command when the event fires.

**log-trap-and-execute**
Generates an RMON log and SNMP trap and executes a batch command when the event fires.

**trap-and-execute**
Generates an SNMP trap and executes a batch command when the event fires.

**argument** *string*
Specifies the batch command argument.

**log**
Generates an RMON log when the event fires.

**trap**
Generates an SNMP trap when the event fires.

**log-and-trap**
Generates an RMON log and SNMP trap when the event fires.

**owner** *event-owner*
Specifies the batch command owner.

# Modes

Global configuration mode

# Usage Guidelines

There are two elements to the Event Group: the event control table and the event log table. The event control table defines the action to be taken when an alarm is reported. Defined events can be found by entering the CLI command **show event**. The event log table collects and stores reported events for retrieval by an RMON application.

The **no** form of the command removes the configured RMON event.

# Examples

The following example configures an RMON event.

```
device(config)# rmon event 1 description 'testing a longer string' trap owner nyc02
```

# rmon history

Configures an RMON history control.

## Syntax

**rmon history** *entry-number* **interface** { **ethernet** *stack-id/slot/port* | **management** *number* } **buckets** *number* **interval** *sampling-interval* **owner** *owner-name*

**no rmon history** *entry-number* **interface** { **ethernet** *stack-id/slot/port* | **management** *number* } **buckets** *number* **interval** *sampling-interval* **owner** *owner-name*

## Command Default

All active ports will generate two history control data entries per active Layer 2 switch port or Layer 3 switch interface.

## Parameters

*entry-number*
Specifies the history number. The value can range from 1 to 65535.

**interface ethernet** *stack-id/slot/port*
Specifies the Ethernet interface to monitor.

**interface management** *number*
Specifies the management interface to monitor.

**buckets** *number*
Specifies the number of buckets. The value can range from 1 to 65535.

**interval** *sampling-interval*
Specifies the sample interval. The value can range from 1 to 3600.

**owner** *owner-name*
Specifies the history owner.

## Modes

Global configuration mode

## Usage Guidelines

An active port is defined as one with a link up. If the link goes down, the two entries are automatically deleted.

Two history entries are generated for each device:

- A sampling of statistics every 30 seconds
- A sampling of statistics every 30 minutes

The history data can be accessed and displayed using any of the popular RMON applications.

The **no** form of the command removes the configured RMON history control.

# Examples

The following example configures the RMON history.

```
device(config)# rmon history 1 interface ethernet 1/1/1 buckets 10 interval 10 owner nyc02
```

# route-only

Enables Ruckus Layer 3 switches to support Layer 2 switching.

## Syntax

**route-only**

**no route-only**

## Command Default

By default, Ruckus Layer 3 switches support Layer 2 switching.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

By default, Ruckus Layer 3 switches support Layer 2 switching. These devices modify the routing protocols that are not supported on the devices. If you want to disable Layer 2 switching, you can do so globally or on individual ports, depending on the version of software your device is running.

Enabling or disabling Layer 2 switching is supported in Layer 3 software images only. Enabling or disabling Layer 2 switching is not supported on virtual interfaces.

Ruckus FCX 6430, FCX 6450, FCX 6430-C12, ICX 6450, and ICX 6610 devices support both the ingress and egress L2 traffic suppression on a route-only port.

Ruckus ICX 7750, ICX 7450, ICX 7250, and ICX 7150 devices support only ingress L2 traffic suppression on a route-only port.

The **no** form of the command enables Layer 2 switching on a Layer 3 switch.

To disable Layer 2 switching only on a specific interface, go to the interface configuration level for that interface, and then configure the command.

## Examples

The following example globally disables Layer 2 switching on a Layer 3 switch.

```
device(config)# route-only
device(config)# exit
device# write memory
device# reload
```

The following example enables Layer 2 switching on a Layer 3 switch.

```
device(config)# no route-only
device(config)# exit
device# write memory
device# reload
```

The following example disables Layer 2 switching on Ethernet interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# route-only
device(config-if-e1000-1/1/1)# end
device# write memory
device# reload
```

# route-precedence

Configures a table that defines the order (precedence) in which multicast routes are selected from the multicast routing table (mRTM) and unicast routing (uRTM) table.

## Syntax

**route-precedence** { [ **mc-non-default** | **none** ] | [ **mc-default** | **none** ] | [ **uc-non-default** | **none** ] | [ **uc-default** | **none** ] }

**no route-precedence**

## Command Default

The default route precedence used to select routes is:

1. A non-default multicast route from the mRTM (**mc-non-default**).
2. A default multicast route from the mRTM (**mc-default**).
3. A non-default unicast route from the uRTM (**uc-non-default**).
4. A default unicast route from the uRTM (**uc-non-default**).

## Parameters

**mc-non-default**
Specifies the precedence for the non-default multicast route table (mRTM).

**none**
Specifies that this type of route is to be ignored. You can specify this option for any of the multicast or unicast route types.

**mc-default**
Specifies the precedence for the multicast routing table (mRTM).

**uc-non-default**
Specifies the precedence for the non-default unicast route table (uRTM).

**uc-default**
Specifies the precedence for the default unicast route table (uRTM).

## Modes

Router PIM configuration mode

## Usage Guidelines

The order in which you place the keywords determines the route precedence.

The **no** form of this command restores the default route precedence settings.

You must configure four parameters indicating the four different route types. If you want to specify that a particular route type is not used, configure the **none** keyword to fill the precedence table.

# Examples

The following example configures a route precedence in which a non-default multicast route has the highest precedence, and a default unicast route has the lowest precedence. The order used to select routes is:

1. A non-default multicast route from the mRTM.
2. A non-default unicast route from the uRTM.
3. A default multicast route from the mRTM.
4. A default unicast route from the uRTM.

```
device(config)# router pim
device(config-pim-router)# route-precedence mc-non-default uc-non-default mc-default uc-default
```

The following example configures a route precedence in which the unicast default route is ignored. The order used to select routes is:

1. A non-default multicast route from the mRTM.
2. A default multicast route from the mRTM.
3. A non-default unicast route from the uRTM.

```
device(config)# router pim
device(config-pim-router)# route-precedence mc-non-default mc-default uc-non-default none
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# route-precedence admin-distance

Configures route precedence so that multicast routes are selected from the best route in the multicast routing table (mRTM) and unicast routing (uRTM) table.

## Syntax

**route-precedence admin-distance**

**no route-precedence admin-distance**

## Command Default

Multicast routes are not selected from the best route in the mRTM and uRTM. Routes are selected based on:

- The route precedence configured using the **route-precedence** command.
- The system route precedence default (if route precedence has not been configured using the **route-precedence** command.

the default route precedence settings.

## Modes

PIM configuration mode

## Usage Guidelines

The **no** form of this command restores the previous route precedence settings.

If the mRTM and the uRTM have routes of equal cost, the route from the mRTM is preferred.

## Examples

The following example configures route precedence so that the best multicast route from the mRTM and uRTM tables is selected.

```
Device(config)#router pim
Device(config-pim-router)#route-precedence admin-distance
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# router bgp

Enables BGP routing.

## Syntax

**router bgp**

**no router bgp**

## Command Default

BGP routing is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

ICX 7150 devices do not support BGP.

The **no** form of the command disables BGP routing.

## Examples

The following example enables BGP routing.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)#
```

# router msdp

Enables multicast source discovery protocol (MSDP) on a router.

## Syntax

**router msdp** [ **vrf** *vrf-name* ]

## Command Default

MSDP is not enabled.

## Parameters

**vrf** *vrf-name*
Specifies a virtual routing and forwarding (VRF) instance.

## Modes

Global configuration mode

## Usage Guidelines

When you configure the **no router msdp vrf** *vrf-name* command, the MSDP configuration is removed only from the specified VRF.

The PIM Sparse Rendezvous Point (RP) is also an MSDP peer.

Devices that run MSDP usually also run BGP. The source address used by the MSDP device is normally configured to be the same source address used by BGP.

All MSDP parameters available for the default router instance are configurable for a VRF-based MSDP instance.

## Examples

The following example enables MSDP.

```
Device(config)# router msdp
```

The following example enables MSDP on a VRF named blue.

```
Device(config)# router msdp vrf blue
```

The following example removes the MSDP configuration only from the VRF named blue.

```
Device(config-msdp-router-vrf-blue)# no router msdp vrf blue
```

# router ospf

Enables and configures the Open Shortest Path First version 2 (OSPFv2) routing protocol.

## Syntax

**router ospf** [ **vrf** *name* ]

**no router ospf**

## Parameters

**vrf** *name*

Specifies a nondefault VRF.

## Modes

Global configuration mode

## Usage Guidelines

Use this command to enable the OSPFv2 routing protocol and enter OSPF router or OSPF router VRFconfiguration mode. OSPFv2 maintains multiple instances of the routing protocol to exchange route information among various VRF instances.

The **no** form of the command deletes all current OSPF configuration and blocks any further OSPFv2 configuration.

## Examples

The following example enables OSPFv2 on a default VRF and enters OSPF VRF router configuration mode.

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)#
```

# router pim

Configures basic global protocol-independent multicast (PIM) Sparse parameters on a device within the PIM Sparse domain and enters PIM-router configuration mode.

## Syntax

**router pim** [ **vrf** *vrf-name* ]

**no router pim** [ **vrf** *vrf-name* ]

## Command Default

PIM Sparse is not configured.

## Parameters

**vrf** *vrf-name*
Specifies a virtual routing and forwarding (VRF) instance.

## Modes

Global configuration mode

Interface configuration mode

## Usage Guidelines

The **no** form of this command disables PIM and removes all configuration for PIM multicast on the device (**router pim** level) only.

Configuring the **no router pim vrf** *vrf-name* command removes all configuration for PIM multicast on the specified VRF.

You do not need to globally enable IP multicast routing when configuring PIM Sparse.

After you enable IP multicast routing and PIM Sparse at the global level, you must enable it on the individual interfaces connected to the PIM Sparse network.

If you configure PIM Sparse on an interface that is on the border of the PIM Sparse domain, you also must also configure the **ip pim border** command on the interface.

You must configure the **bsr-candidate ethernet** command to identify an interface on at least one device as a candidate PIM Sparse Bootstrap router (BSR) and candidate PIM Sparse Rendezvous Point (RP).

You can configure the **rp-address** command to explicitly identify an RP, including an ACL-based RP, by its IP address instead of having it identified by the RP election process.

Entering the **router pim vrf** command to enable PIM does not require a software reload.

All PIM parameters available for the default router instance are configurable for a VRF-based PIM instance.

# Examples

This example configures basic global PIM Sparse parameters.

```
device(config)# router pim
```

This example configures PIM Sparse on a VRF named blue.

```
device(config)# router pim blue
```

# router rip

Enables Routing Information Protocol (RIP) globally on the device. Does not enable RIP at the interface level.

## Syntax

**router rip**

**no router rip**

## Command Default

By default, RIP is not enabled on the device.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables RIP on the device.

Once you have enabled RIP on the device, you must also configure RIP on each RIP interface. Refer to the **ip rip** command for more information.

## Examples

The following example enables RIP on a device.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)#
```

# router vrrp

Globally enables Virtual Router Redundancy Protocol (VRRP).

## Syntax

**router vrrp**

**no router vrrp**

## Command Default

VRRP is not globally enabled.

## Modes

Global configuration mode

## Usage Guidelines

After globally enabling VRRP, the command prompt does not change. Nearly all subsequent VRRP configuration is performed at the interface level, but VRRP must be enabled globally before configuring VRRP instances.

The **no router vrrp** command disables VRRP globally.

**NOTE**
Only 16 VRRP instances are configurable on the ICX 7150 device.

## Examples

The following example globally enables VRRP and enters interface configuration mode.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/5
```

# router vrrp-extended

Globally enables Virtual Router Redundancy Protocol Extended (VRRP-E) and enters VRRP-E router configuration mode.

## Syntax

**router vrrp-extended**

**no router vrrp-extended**

## Command Default

VRRP-E is not globally enabled.

## Modes

Global configuration mode

## Usage Guidelines

After globally enabling VRRP-E, nearly all subsequent VRRP-E configuration is performed at the interface level. VRRP-E must be enabled globally before configuring VRRP-E instances.

The **no router vrrp-extended** command globally disables VRRP-E.

**NOTE**
Only 16 VRRP instances are configurable on the ICX 7150 device.

## Examples

The following example globally enables VRRP-E and enters interface configuration mode.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# interface ethernet 1/1/5
device(config-if-e1000-1/1/5)# ip address 10.53.5.3/24
device(config-if-e1000-1/1/5)# ip vrrp-extended vrid 1
device(config-if-e1000-1/1/5-vrid-1)# backup priority 110
device(config-if-e1000-1/1/5-vrid-1)# version 2
device(config-if-e1000-1/1/5-vrid-1)# ip-address 10.53.5.254
device(config-if-e1000-1/1/5-vrid-1)# activate
VRRP-E router 1 for this interface is activating
```

# router vsrp

Enables the Virtual Switch Redundancy Protocol (VSRP) on Layer 2 or Layer 3 switches.

## Syntax

**router vsrp**

**no router vsrp**

## Command Default

By default, VSRP is enabled on Layer 2 and Layer 3 switches.

## Modes

Global configuration mode

## Usage Guidelines

On a Layer 3 switch, if you want to use VRRP or VRRP-E for Layer 3 redundancy instead of VSRP, you must disable VSRP first. Because VRRP and VRRP-E do not apply to Layer 2 switches, there is no need to disable VSRP and there is no command to do so. VSRP is always enabled on Layer 2 switches.

The **no** form of the command disables VSRP.

## Examples

The following example shows how to disable VSRP and then enable it.

```
device(config)# no router vsrp
device(config)# router vsrp
```

# router-interface

Attaches a router interface to a Layer 2 VLAN.

## Syntax

**router-interface ve** *num*

**no router-interface ve** *num*

## Command Default

A router interface is not configured.

## Parameters

**ve** *num*

Specifies a virtual router interface number.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the router interface from the VLAN.

## Examples

The following example shows how to attach the router interface to a Layer 2 VLAN.

```
device(config)# vlan 1 by port
device(config-vlan-1)# untagged ethernet 1/1/1
device(config-vlan-1)# tagged ethernet 1/1/8
device(config-vlan-1)# router-interface ve 1
```

# rpf-mode

Enables strict or loose unicast Reverse Path Forwarding (uRPF) mode on FastIron ICX devices.

## Syntax

**rpf-mode** [ **strict** [ **urpf-exclude-default** ] | **loose** [**urpf-exclude-default** ] ]

**no rpf-mode** [ **strict** [ **urpf-exclude-default** ] | **loose** [**urpf-exclude-default** ] ]

## Command Default

uRPF mode is not enabled.

## Parameters

**strict**
Specifies uRPF strict mode.

**loose**
Specifies uRPF loose mode. This mode allows all packets to pass the uRPF check.

**urpf-exclude-default**
Excludes the default route for uRPF source IP lookup.

## Modes

Interface configuration mode

## Usage Guidelines

You must enable uRPF at the global level before enabling the mode (strict or loose). This command is applicable only to the Layer 3 physical interface and Layer 3 VE interfaces.

The **loose** option allows all packets to pass through. Choose the **loose** option along with the **urp-exclude-default** option to subject the packets to uRPF check.

The **no** form of the command disables uRPF mode.

## Examples

The following example sets the Reverse Path Forwarding mode to strict mode.

```
device(config)# interface ethernet 1/1/3
device(config-if-e1/1/3)# rpf-mode strict
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# rp-address

Configures a device interface as a rendezvous point (RP).

## Syntax

**rp-address** { *ip-address* | *ipv6-address* } *acl_name_or_id*

**no rp-address** { *ip-address* | *ipv6-address* }

## Command Default

The RP is selected by the PIM Sparse protocol's RP election process.

## Parameters

*ip-address*
Specifies the IP address of the RP.

*ipv6-address*
Specifies the IPv6 address of the RP.

*acl_name_or_id*
Specifies the name or ID of the ACL that specifies which multicast groups use the RP.

## Modes

Router PIM configuration mode

VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the default and the RP is selected by the RP election process.

Devices in the PIM Sparse domain use the specified RP and ignore group-to-RP mappings received from the bootstrap router (BSR).

The RP is the meeting point for PIM Sparse sources and receivers. A PIM Sparse domain can have multiple RPs, but each PIM Sparse multicast group address can have only one active RP. PIM Sparse routers learn the addresses of RPs and the groups for which they are responsible from messages that the BSR sends to each of the PIM Sparse routers.

**NOTE**
Specify the same IP or IPv6 address as the RP on all PIM Sparse devices within the PIM Sparse domain. Make sure the device is on the backbone or is otherwise well connected to the rest of the network.

# Examples

This example configures the device interface at IP address 207.95.7.1 as the RP for the PIM Sparse domain.

```
device(config)# router pim
device(config-pim-router)# rp-address 207.95.7.1
```

This example configures an ACL named acl1 to specify which multicast groups use the RP.

```
device(config)# router pim
device(config-pim-router)# rp-address 130.1.1.1 acl1
```

This example configures an RP for a VRF named blue.

```
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# rp-address 31::207
```

# rp-adv-interval

Configures the interval at which the candidate rendezvous point (RP) configured on the device sends candidate-RP advertisement messages to the bootstrap router (BSR).

## Syntax

**rp-adv-interval** *seconds*

**no rp-adv-interval** *seconds*

## Command Default

The device sends candidate-RP advertisement messages every 60 seconds.

## Parameters

*seconds*

Specifies the interval, in seconds, between advertisement messages. The range is 10 through 65535 seconds. The default is 60 seconds.

## Modes

PIM router configuration mode

PIM router VRF configuration mode

## Usage Guidelines

The **no** form of this command restores the candidate-RP advertisement-message interval to 60 seconds.

## Examples

The following example configures the candidate-RP advertisement-message interval to 90 seconds.

```
device(config)# router pim
device(config-pim-router)# rp-adv-interval 90
```

The following example configures, on a VRF named blue, the candidate-RP advertisement-message interval to 90 seconds.

```
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# rp-adv-interval 90
```

# rp-candidate

Configures a device as a candidate rendezvous point (RP) for all multicast groups with the prefix 224.0.0.0/4, by default, and explicitly adds or deletes groups with other prefixes.

## Syntax

**rp-candidate** { **ethernet** *stackid* **/** *slot* **/** *portnum* | **loopback** *num* | **ve** *num* | **tunnel** *num* }

**rp-candidate** {**add** | **delete** } *group-addr mask-bits*

**no rp-candidate** { **ethernet** *stackid* **/** *slot* **/** *portnum* | **loopback** *num* | **ve** *num* | **tunnel** *num* }

**no rp-candidate** {**add** | **delete** } *group-addr mask-bits*

## Command Default

The PIM router is not available for selection as an RP.

## Parameters

**ethernet** *stackid***/***slot***/***portnum*

Specifies a physical interface for the candidate RP. On standalone devices specify the interface ID in the format slot/port-id; on stacked devices you must also specify the stack ID, in the format stack-id/slot/port-id.

**loopback** *num*

Specifies a loopback interface for the candidate RP.

**ve** *num*

Specifies a virtual interface for the candidate RP.

**tunnel** *num*

Specifies a GRE tunnel interface for the candidate RP.

**add**

Specifies adding a group address or range of group addresses to the default group configured by the those the device is the candidate RP for by default, that is, groups with the prefix 224.0.0.0/4.

**delete**

Specifies deleting a group address or range of group addresses, that were added using the **add** keyword.

*group-addr mask-bits*

Specifies the group address and the number of significant bits in the subnet mask.

## Modes

Router PIM configuration mode

## Usage Guidelines

The **no rp-candidate** command makes the PIM router cease to act as a candidate RP.

The **no rp-candidate add** command deletes a group address or range of group addresses that were added using the **add** keyword.

Configuring the **rp-candidate** command on an Ethernet, loopback, virtual, or tunnel interface, configures the device as a candidate RP for all multicast groups with the prefix 224.0.0.0/4, by default. You can configure the **rp-candidate add** command to add to those a group address or range of group addresses. You can configure the **rp-candidate delete** command to delete a group address or range of group addresses that were added to the default addresses.

**NOTE**
You cannot delete the default group prefix.

The RP is the meeting point for PIM Sparse sources and receivers. A PIM Sparse domain can have multiple RPs, but each PIM Sparse multicast group address can have only one active RP. PIM Sparse routers learn the addresses of RPs and the groups for which they are responsible from messages that the bootstrap router (BSR) sends to each of the PIM Sparse routers.

Although you can configure the device as only a candidate BSR or an RP, it is recommended that you configure the same interface on the same device as both a BSR and an RP.

**NOTE**
Specify the same IPv6 address as the RP on all IPv6 PIM Sparse routers within the IPv6 PIM Sparse domain. Make sure the device is on the backbone or is otherwise well connected to the rest of the network. You can configure the **rp-address** command to specify the RP address.

# Examples

This example configures a physical device as a candidate RP.

```
device(config)# router pim
device(config-pim-router)# rp-candidate ethernet 1/2/2
```

This example uses a loopback interface to configure a device as a candidate RP.

```
device(config)# router pim
device(config-pim-router)# rp-candidate loopback 1
```

This example uses a virtual interface to configure a device as a candidate RP.

```
device(config)# router pim
device(config-pim-router)# rp-candidate ve 120
```

This example configures an address group to the devices for which it is a candidate RP.

```
device(config)# router pim
device(config-pim-router)# rp-candidate add 224.126.0.0 16
```

This example deletes an address group from the devices for which it is a candidate RP.

```
device(config)# router pim
device(config-pim-router)# rp-candidate delete 224.126.22.0 24
```

# History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was modified to add the **tunnel** keyword. |

# rp-embedded

Configures embedded-rendezvous point (RP) support on PIM devices.

## Syntax

**rp-embedded**

**no rp-embedded**

## Command Default

Embedded RP support is enabled.

## Modes

PIM router configuration mode

PIM router VRF configuration mode

## Usage Guidelines

The **no** form of this command disables embedded RP support.

## Examples

This example disables embedded RP support.

```
Device(config)# ipv6 router pim
Device(config-ipv6-pim-router)#no rp-embedded
```

This example disables embedded RP support on a VRF named blue.

```
Device(config)#ipv6 router pim vrf blue
Device(config-ipv6-pim-router-vrf-blue)#no rp-embedded
```

# rsakeypair (PKI)

Specifies which RSA keypair to use during enrollment.

## Syntax

**rsakeypair** { **key-label** *key_name* }

**no rsakeypair** { **key-label** *key_name* }

## Command Default

## Parameters

**key-label** *key_name*
Designates the RSA key to be used for enrollment.

## Modes

PKI trustpoint configuration sub-mode.

## Usage Guidelines

The no form of the command removes the configuration.

The **crypto key generate rsa** command is used to create and name an RSA keypair.

## Examples

The following example...

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# rspan destination

Configures a Remote Switched Port Analyzer (RSPAN) destination port for port mirroring.

## Syntax

**rspan destination** { **ethernet** *unit/slot/port* | **lag** *lag-id* }

**no rspan destination** { **ethernet** *unit/slot/port* | **lag** *lag-id* }

## Command Default

No RSPAN destination port is configured.

## Parameters

**ethernet** *unit/slot/port*

Specifies the Ethernet interface and the interface ID in the unit/slot/port format.

**lag** *lag-id*

Specifies the LAG virtual interface.

## Modes

RSPAN configuration mode

## Usage Guidelines

The configured VLAN must not be a user VLAN. The destination Interface must be a member port for the RSPAN VLAN.

The **no** form of the command deletes the port mirroring destination port for the specified interface.

## Examples

The following example configures Ethernet interface 1/1/2 as an RSPAN destination port for a switch.

```
device# configure terminal
device(config)# rspan-vlan 4000
device(config-rspan-vlan)# tagged ethernet 1/1/2
device(config-rspan-vlan)# rspan destination ethernet 1/1/2
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# rspan source

Configures a Remote Switched Port Analyzer (RSPAN) source port and properties for port mirroring.

## Syntax

**rspan source** { **monitor-both** | **monitor-in** | **monitor-out** } { **ethernet** *unit/slot/port* | **lag** *lag-id* }

**no rspan source** { **monitor-both** | **monitor-in** | **monitor-out** } { **ethernet** *unit/slot/port* | **lag** *lag-id* }

## Command Default

No RSPAN source port is configured.

## Parameters

**monitor-both**
Specifies ingress and egress traffic.

**monitor-in**
Specifies ingress traffic only.

**monitor-out**
Specifies egress traffic only.

**ethernet** *unit/slot/port*
Specifies the Ethernet interface and the interface ID in the unit/slot/port format.

**lag** *lag-id*
Specifies the LAG virtual interface.

## Modes

RSPAN configuration mode

## Usage Guidelines

The configured VLAN must not be a user VLAN. This command can be successfully executed only if an RSPAN destination port is configured. RSPAN must be configured on all the switches participating in the RSPAN session.

The **no** form of the command deletes the port mirroring source port for the specified interface.

## Examples

The following example configures an Ethernet interface as an RSPAN source port and specifies that ingress traffic is monitored for a device.

```
device# configure terminal
device(config)# rspan-vlan 4000
device(config-rspan-vlan)# tagged ethernet 1/1/2
device(config-rspan-vlan)# rspan source monitor-in ethernet 1/1/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# rspan-vlan

Configures the VLAN to support Remote Switched Port Analyzer (RSPAN) traffic analysis.

## Syntax

**rspan-vlan** *vlan-id*

**no rspan-vlan** *vlan-id*

## Command Default

RSPAN traffic analysis is not supported for the VLAN.

## Parameters

*vlan-id*

Specifies the VLAN ID.

## Modes

Global configuration mode

## Usage Guidelines

The VLAN must not be a user VLAN.

The **no** form of the command deletes RSPAN for the VLAN.

## Examples

The following example configures VLAN 4000 to support RSPAN.

```
device# configure terminal
device(config)# rspan-vlan 4000
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# sa-filter

Configures filters for incoming and outgoing Source-Active (SA) messages from and to multicast source discovery protocol (MSDP) neighbors.

## Syntax

**sa-filter** { **in** | **out** } *ip-addr* [ **route-map** *map-tag* [ **rp-route-map** *rp-map-tag* ] ]

**no sa-filter** { **in** | **out** } *ip-addr* [ **route-map** *map-tag* [ **rp-route-map** *rp-map-tag* ] ]

**sa-filter originate** [ **route-map** *map-tag* ]

**no sa-filter originate** [ **route-map** *map-tag* ]

## Command Default

Source-Active filters are not configured.

## Parameters

**in**
Specifies filtering incoming SA messages.

**out**
Specifies filtering self-originated and forwarded outbound SA messages.

*ip-addr*
specifies the IP address of the MSDP neighbor that the filtered SA messages are sent to of received from.

**originate**
Specifies filtering self-originated outbound SA messages.

**route-map** *map-tag*
Specifies a route map. The device applies the filter to source-group pairs that match the route map.

**rp-route-map** *rp-map-tag*
Specifies a route map to use for filtering based on Rendezvous Point (RP) address. Use this parameter if you want to filter SA messages based on their originating RP.

## Modes

MSDP VRF configuration mode

Router MSDP configuration mode

## Usage Guidelines

The default filter action is deny. If you want to permit some source-group pairs, use a route map. A permit action in the route map allows the device to advertise the matching source-group pairs. A deny action in the route map drops the source-group pairs from advertisements.

The **no** form of this command removes the SA filters.

# Examples

The following example configures extended access-control lists (ACLs) to be used in the route map definition and use them to configure a route map that denies source-group with source address 10.x.x.x and any group address, while permitting everything else.

```
device(config)# access-list 123 permit ip 10.0.0.0 0.255.255.255 any
device(config)# access-list 125 permit ip any any
device(config)# route-map msdp_map deny 1
device(config-routemap msdp_map)# match ip address 123
device(config-routemap msdp_map)# exit
device(config)# route-map msdp_map permit 2
device(config-routemap msdp_map)# match ip address 125
device(config-routemap msdp_map)# exit
```

The following example configures a filter that filters self-originated outbound SA messages on a route map.

```
device(config)# router msdp
device(config-msdp-router)# sa-filter originate route-map msdp_map
```

The following example configures an SA filter on a VRF.

```
device(config)# router msdp vrf blue
device(config-msdp-router-vrf blue)# sa-filter in 2.2.2.99
device(config-msdp-router-vrf blue)# sa-filter in 2.2.2.97 route-map msdp_map
device(config-msdp-router-vrf blue)# sa-filter in 2.2.2.96 route-map msdp2_map rproute-map msdp2_rp_map
```

# save-current-values

Configures a backup to save the VSRP timer values received from the master instead of the timer values configured on the backup.

## Syntax

**save-current-values**

**no save-current-values**

## Command Default

By default, the backups always use the value of the timers received from the master.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

Saving the current timer values instead of the configured ones helps ensure consistent timer usage for all the VRID devices.

The **no** form of the command disables saving the timer values from the master.

## Examples

The following example shows how to configure a backup to save the VSRP timer values received from the master instead of the timer values configured on the backup.

```
device(config)# vlan 200
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# save-current-values
```

# scp

Copies a license file from an SCP-enabled client to the license database of the device.

## Syntax

**scp** *license_file_on_hostuser@IP_address:* **license:***unit_id*

## Command Default

By default, the command is not enabled.

## Parameters

**license_file_on_hostuser@IP_address:**
Specifies the filename of the license file at the specified IP address.

**license**
Specifies the keyword license to be used.

*unit_id*
Indicates the specific unit you want to copy the software license file to. The *unit-id* can be from 1 through 12.

## Usage Guidelines

The *unit_id* parameter is used on Ruckus ICX devices when copying a license file from an SCP-enabled client to a specific unit id.

## Examples

The following example copies the license file from an SCP-enabled client to the license database of a specific unit on the Ruckus ICX devices. In the example the license is copied to unit 3.

```
device# scp license.xml terry@10.20.91.39:license:3
```

## History

| Release version | Command history |
|---|---|
| 07.2.00 | This command was introduced. |
| 05.0.00 | This command was introduced. |

# secure-login

Configures Web Authentication to use secure (HTTPS) or non-secure (HTTP) login and logout pages.

## Syntax

**secure-login**

**no secure-login**

## Command Default

Web Authentication uses secure (HTTPS) login and logout pages.

## Modes

Web Authentication configuration mode

## Usage Guidelines

The **no** form of the command changes the setting to non-secure (HTTP) mode.

## Examples

The following example configures Web Authentication to use non-secure (HTTP) login.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# no secure-login
```

# secure-mac-address

Configures secure MAC addresses on tagged and untagged interfaces.

## Syntax

**secure-mac-address** *mac-address* [ *vlan-id* ]

**no secure-mac-address** *mac-address* [ *vlan-id* ]

## Command Default

Secure MAC addresses are not configured.

## Parameters

*mac-address*

Specifies the MAC address.

*vlan-id*

Specifies the VLAN ID.

## Modes

Port security interface configuration mode

## Usage Guidelines

When specifying a secure MAC address on a tagged interface, you must also specify the VLAN ID.

**NOTE**
If MAC port security is enabled on a port and you change the VLAN membership of the port, make sure that you also change the VLAN ID specified in the **secure-mac-address** configuration statement for the port.

When a secure MAC address is applied to a tagged port, the VLAN ID is generated for both tagged and untagged ports. When you display the configuration, you see an entry for the secure MAC addresses.

The **no** form of the command removes the configured secure MAC address.

## Examples

The following example shows how to specify a secure MAC address on an untagged interface.

```
device(config)# interface ethernet 1/7/11
device(config-if-e1000-1/7/11)# port security
device(config-port-security-e1000-1/7/11)# secure-mac-address 0000.0018.747C
```

The following example shows how to specify a secure MAC address on a tagged interface.

```
device(config)# interface ethernet 1/7/11
device(config-if-e1000-1/7/11)# port security
device(config-port-security-e1000-1/7/11)# secure-mac-address 0000.0018.747C 2
```

# server (NTP)

Configures the device in client mode and specifies the NTP servers to synchronize the system clock.

## Syntax

**server** { *ipv4-address* | *ipv6-address* } [ **maxpoll** *interval* ] [ **minpoll** *interval* ] [ **version** *version-number* ] [ **key** *key-id* ] [ **burst** ]

**no server** { *ipv4-address* | *ipv6-address* } [ **maxpoll** *interval* ] [ **minpoll** *interval* ] [ **version** *version-number* ] [ **key** *key-id* ] [ **burst** ]

## Parameters

*ipv4-address*
Specifies the IPv4 address of the server that provides the clock synchronization.

*ipv6-address*
Specifies the IPv6 address of the server that provides the clock synchronization.

**version** *version-number*
Specifies the Network Time Protocol (NTP) version number. Valid values are 3 and 4. The default value is 4.

**key** *key-id*
Specifies the authentication key range. The value can range from 1 to 65535.

**minpoll** *interval*
Specifies the shortest polling interval. The range is from 4 through 17. The default is 6. The interval argument is a power of 2 (4=16, 5=32, 6=64, 7=128, 8=256, 9=512, and so on).

**maxpoll** *interval*
Specifies the longest polling interval. The range is from 4 through 17. The default is 10. The interval argument is a power of 2 (4=16, 5=32, 6=64, 7=128, 8=256, 9=512, and so on).

**burst**
Sends a burst of packets to the server at each polling interval.

## Modes

NTP configuration mode

## Usage Guidelines

A maximum of eight NTP servers can be configured.

The **no** form of the command removes the NTP server configuration.

## Examples

The following example configures the NTP server.

```
device(config)# ntp
device(config-ntp)# server 10.1.1.1 key 23 maxpoll 15 minpoll 8 version 3 burst
```

# service local-user-protection

Prevents unauthorized deletion or modification of a user account.

## Syntax

**service local-user-protection**

**no service local-user-protection**

## Command Default

The user account can be deleted or modified without any authentication; that is, user account security is disabled.

## Modes

Global configuration mode

## Usage Guidelines

This command allows for the deletion of user accounts or changing the password or privilege level of the user (using the **username** command) only upon successful validation of the existing user password.

If the command is enabled and you try to delete or modify a user account using the **username**, you will be prompted for confirmation to proceed. Upon confirmation, you will be prompted to provide the existing password. The attempt to modify or delete a user account is successful only if the correct password is entered.

The **no** form of the command disables user account security; the deletion or modification of the user account without any authentication is allowed.

## Examples

The following example permits the modification of the user account password only after providing the existing password.

```
device(config)# username user1 password xpassx
device(config)# service local-user-protection
device(config)# username user1 password ypasswordy
User already exists. Do you want to modify: (enter 'y' or 'n'): y
To modify or remove user, enter current password: ******
```

The following example prevents unauthorized modification of the user account password.

```
device(config)# username user1 password ypasswordy
device(config)# service local-user-protection
device(config)# username user1 password zpassz
User already exists. Do you want to modify: (enter 'y' or 'n'): y
To modify or remove user, enter current password: ****
Error: Current password doesn't match. Access denied
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# service password-encryption

Configures the password encryption service to encrypt the passwords using different encryption methods.

## Syntax

**service password-encryption** { **sha1** | **sha256** }

**no service password-encryption** { **sha1** | **sha256** }

## Command Default

The user account password is encrypted using the MD5 encryption type.

## Parameters

**sha1**

Encrypts system passwords using the SHA 1 encryption type.

**sha256**

Encrypts system passwords using the SHA 256 encryption type.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command reverts the password encryption service type to MD5.

## Examples

The following example specifies the user account password to be encrypted using SHA 1.

```
device(config)# service password-encryption sha1
Warning: Moving to higher password-encryption type, Do you want to continue(y/n)? (enter 'y' or 'n'): y
```

The following example specifies the user account password to be encrypted using SHA 256.

```
device(config)# service password-encryption sha256
Warning: Moving to higher password-encryption type, Do you want to continue(y/n)? (enter 'y' or 'n'): y
```

The following example reverts the password encryption service type to MD5.

```
device(config)# no service password-encryption sha1
Warning: Moving to lower password-encryption type, Do you want to continue(y/n)? (enter 'y' or 'n'): y
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# set interface null0

Drops traffic when the null0 statement becomes the active setting as determined by the route-hop selection process.

## Syntax

**set interface null0**

**no set interface null0**

## Command Default

The configuration to direct the traffic to the null0 interface is not configured.

## Modes

Route map configuration mode

## Usage Guidelines

This command sends the traffic to the null0 interface, which is the same as dropping the traffic.

The **no** form of this command deletes the matching filter from the ACL.

## Examples

The following example configures the PBR policy to send all traffic from 192.168.1.204/32 to the null interface, thus dropping the traffic instead of forwarding it.

```
device(config)# access-list 56 permit 192.168.1.204 0.0.0.0

device(config)# route-map file-13 permit 56
device(config-routemap file-13)# match ip address 56
device(config-routemap file-13)# set interface null0
device(config-routemap file-13)# exit
```

# set next-hop-ip-tunnel

Configures an IPsec or GRE tunnel interface as the next hop of a PBR route map.

## Syntax

**set next-hop-ip-tunnel** *tunnel-id*

**no set next-hop-ip-tunnel** *tunnel-id*

## Command Default

The next hop is not set to a tunnel interface.

## Parameters

*tunnel-id*

Specifies the ID of the tunnel interface.

## Modes

Route map configuration mode

## Usage Guidelines

When PBR is used to map IP traffic into a GRE tunnel or IPsec tunnel, the VRF of the tunnel interface is considered as the egress VRF interface.

The **no** form of the command removes the tunnel interface as the PBR next hop.

## Examples

The following example configures tunnel interface 1 as the PBR next hop.

```
device# interface tunnel 1
device(config-tnif-1)# vrf forwarding blue
device(config-tnif-1)# tunnel source ethernet 1/1/1
device(config-tnif-1)# tunnel destination 10.2.2.1
device(config-tnif-1)# tunnel mode ipsec ipv4
device(config-tnif-1)# tunnel protection ipsec profile prof-blue
device(config-tnif-1)# ip address 10.4.4.4/24
device(config-tnif-1)# exit
device(config)# access-list 99 permit 10.157.23.0 0.0.0.255
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match ip address 99
device(config-routemap test-route)# set next-hop-ip-tunnel 1
device(config-routemap test-route)# end
device(config)# interface ethernet 1/1/3
device(config-if-e1000-1/1/3)# vrf forwarding blue
device(config-if-e1000-1/1/3)# ip policy route-map test-route
device(config-if-e1000-1/1/3)# end
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# scale-timer

Changes the timer scale, the value used by the software to calculate all VSRP timers.

## Syntax

**scale-timer** *number*

**no scale-timer** *number*

## Command Default

By default, the timer scale is set to 1.

## Parameters

*number*

Specifies the multiplier factor for the timer. The range for the timer is from 1 through 10.

## Modes

Global configuration mode

## Usage Guidelines

Increasing the timer scale value decreases the length of all the VSRP timers equally, without changing the ratio of one timer to another.

To achieve sub-second failover times, you can shorten the duration of all scale timers for VSRP, VRRP, and VRRP-E by adjusting the timer scale. The timer scale is a value used by the software to calculate the timers. If you increase the timer scale, each timer value is divided by the scale value. Using the timer scale to adjust timer values enables you to easily change all the timers while preserving the ratios among their values. For example, if you set the timer scale to 2, all VSRP, VRRP, and VRRP-E timer values will be divided by 2. Here is an example:

| Timer | Timer scale | Timer value |
|---|---|---|
| Hello interval | 1 | 1 second |
| | 2 | 0.5 seconds |
| Dead interval | 1 | 3 seconds |
| | 2 | 1.5 seconds |
| Backup Hello interval | 1 | 60 seconds |
| | 2 | 30 seconds |
| Hold-down interval | 1 | 3 seconds |
| | 2 | 1.5 seconds |

The **no** form of the command sets the multiplier to 1.

# Examples

The following example shows how to set the scale timer to 2.

```
device(config)# scale-timer 2
```

# scale-timer

Configures a scale time factor that increases the timing sensitivity across all configured and default Virtual Router Redundancy Protocol Extended (VRRP-E) timers.

## Syntax

**scale-timer vrrp-extended** *scale-factor*

**no scale-timer vrrp-extended** *scale-factor*

## Command Default

VRRP timers are not scaled.

## Parameters

**vrrp-extended**

A scale time factor can be configured for VRRP-E timers.

*scale-factor*

A number representing the scale of the division of a VRRP-E configured interval timer or the default interval timer. Valid values are in a range from 1 through 10. The default value is 1.

## Modes

VRRP-E router configuration mode

## Usage Guidelines

Configuring the VRRP-E scale timer is supported only in VRRP-E sessions. When a scaling value is configured, the existing timer values are divided by the scaling value. For example: a value of 10 divides the timers by a factor of 10, allowing the default dead interval to be set to 300 ms. Using timer scaling, VRRP-E subsecond convergence is possible if a master VRRP device fails.

**NOTE**

Increased timing sensitivity as a result of this configuration could cause protocol flapping during periods of network congestion.

## Examples

The following example scales all VRRP-E timers by a factor of 10.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# scale-timer vrrp-extended 10
```

# scheduler-profile

Attaches a scheduler profile to one or more ports.

## Syntax

**scheduler-profile** *profile-name*

**no scheduler-profile** *profile-name*

## Command Default

A scheduler profile is not attached to a port.

## Parameters

*profile-name*

Specifies the name of the scheduler profile to be attached to the port.

## Modes

Interface mode

Multiple-interface mode

## Usage Guidelines

The **no** form of this command removes the scheduler profile from the port or ports.

You must configure a user scheduler profile before you can attach it to a port.

Only one scheduler profile at a time can be attached to any port. You can attach a scheduler profile to more than one port.

## Examples

The following example attaches a scheduler profile named user1 to a port.

```
Device(config-if-e10000-1/1/1)# scheduler-profile user1
```

The following example attaches a scheduler profile named user2 to multiple ports.

```
Device(config-mif-1/1/2-1/1/16)# scheduler-profile user2
```

The following example removes a scheduler profile named user2 from multiple ports.

```
Device(config-mif-1/1/2-1/1/16)# no scheduler-profile user2
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# sequence (permit | deny in extended IPv4 ACLs)

Inserts filtering rules in IPv4 extended named or numbered ACLs.

## Syntax

Use the following syntax to define a TCP or UDP rule:

[ **sequence** *seq-num* ] { **deny** | **permit** } { **tcp** | **udp** } { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ *source-comparison-operators* ] { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ **established** ] [ *destination-comparison-operators* ] [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define an ICMP rule:

[ **sequence** *seq-num* ] { **deny** | **permit** } **icmp** { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ *icmp-num* | *icmp-type* ] [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define a rule for protocols other than TCP, UDP, or ICMP:

[ **sequence** *seq-num* ] { **deny** | **permit** } *ip-protocol* { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } { *D_IPaddress* [ *mask* ] | **host** *D_IPaddress* | **any** } [ **precedence** { *precedence-name* | *precedence-value* } ] [ **tos** { *tos-name* | *tos-value* } ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **802.1p-and-internal-marking** *priority-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

**no sequence** *seq-num*

## Parameters

**sequence**

(Optional) Enables you to assign a sequence number to the rule.

*seq-num*

Valid values range from 1 through 65000.

**deny**

Specifies rules to deny traffic.

**permit**

Specifies rules to permit traffic.

*ip-protocol*

Specifies the type of IPv4 packet to filter. You can either specify a protocol number (from 0 through 255) or a supported protocol name. For a complete list of protocols, type **?** after **permit** or **deny**. Supported protocols include:

- **icmp**—Internet Control Message Protocol
- **igmp**—Internet Group Management Protocol
- **igrp**—Internet Gateway Routing Protocol

- **ip**—any IPv4 protocol
- **ospf**—Open Shortest Path First
- **tcp**—Transmission Control Protocol
- **udp**—User Datagram Protocol

*S_IPaddress*

Specifies a source address for which you want to filter the subnet.

*mask*

Defines a mask, whose effect is to specify a subnet that includes the source address that you specified. For options to specify the mask, see the Usage Guidelines.

**host**

Specifies the source as a host.

*S_IPaddress*

Specifies the source address of the host.

**any**

Specifies all source addresses.

*source-comparison-operators* and *destination-comparison-operators*

If you specified **tcp** or **udp**, the following optional operators are available:

**eq**

Specifies the address is equal to the port name or number you enter after **eq**.

**gt**

Specifies port numbers that are equal to or greater than the port number or that are equal to or greater than the numeric equivalent of the port name you enter after **gt**.

**lt**

Specifies port numbers that are equal to or less than the port number or that are equal to or less than the numeric equivalent of the port name you enter after **lt**.

**neq**

Specifies all port numbers except the port number or port name you enter after **neq**.

**range**

Specifies all port numbers that are between the first port name or number and the second name or number you enter following the **range** keyword. Enter the range as two values separated by a space. The first port number in the range must be less than the last number in the range. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: 23 53 .

*D_IPaddress*

Specifies a destination address for which you want to filter the subnet.

*mask*

Defines a subnet mask that includes the destination address that you specified. For mask options, refer to the Usage Guidelines.

**host**

Specifies a host as destination.

*D_IPaddress*

Specifies the destination address of the host.

**any**
Specifies all destination addresses.

**established**
(For TCP rules only) Filter packets that have the Acknowledgment (ACK) or Reset (RST) flag set. This policy applies only to established TCP sessions, not to new sessions.

*icmp-num* | *icmp-type*
(For ICMP only) Specifies a named or numbered message type.

*icmp-num*
Specifies a numbered message type. Use this format if the rule also needs to include **precedence**, **tos** , one of the DSCP options, one of the 802.1p options, **internal-priority-marking** , or **traffic-policy**.

**any-icmp-type**
Specifies any ICMP type.

**echo**
Specifies an echo request (ping).

**echo-reply**
Specifies an echo reply.

**information-request**
Specifies an information request.

**mask-reply**
Specifies an address mask reply.

**mask-request**
Specifies an address mask request.

**parameter-problem**
Specifies a parameter problem.

**redirect**
Specifies a redirect message.

**source-quench**
Specifies a relieve congestion message.

**time-exceeded**
Specifies a time exceeded message.

**timestamp-reply**
Specifies a timestamp reply.

**timestamp-request**
Specifies a timestamp request.

**unreachable**
Specifies a destination-unreachable message.

**precedence** { *precedence-name* | *precedence-value* }
Specifies a *precedence-name* or corresponding *precedence-value*, as follows:

**0** or **routine**
Specifies routine precedence.

**1** or **priority**
Specifies priority precedence.

**2** or **immediate**
Specifies immediate precedence.

**3** or **flash**
Specifies flash precedence.

**4** or **flash-override**
Specifies flash-override precedence.

**5** or **critical**
Specifies critical precedence.

**6** or **internet**
Specifies internetwork control precedence.

**7** or **network**
Specifies network control precedence.

**tos** { *tos-name* | *tos-value* }
Specifies a type of service (ToS). Enter either a supported *tos-name* or the equivalent *tos-value*.

**0** or **normal**
Specifies normal ToS.

**1** or **min-monetary-cost**
Specifies min monetary cost ToS.

**2** or **max-reliability**
Specifies max reliability ToS.

**4** or **max-throughput**
Specifies max throughput ToS.

**8** or **min-delay**
Specifies min-delay ToS.

**dscp-matching** *dscp-value*
Filters by DSCP value. Values range from 0 through 63.

**dscp-marking** *dscp-value*
Assigns the DSCP value that you specify to the packet. Values range from 0 through 63.

**802.1p-priority-matching** *802.1p-value*
Filters by 802.1p priority, for rate limiting. Values range from 0 through 7.

**802.1p-priority-marking** *802.1p-value*
Assigns the 802.1p value that you specify to the packet. Values range from 0 through 7.

**internal-priority-marking** *queuing-priority*
Assigns the internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**802.1p-and-internal-marking** *priority-value*
Assigns the identical 802.1p value and internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**traffic-policy** *name*

Enables the device to limit the rate of inbound traffic and to count the packets and bytes per packet to which ACL permit or deny clauses are applied. For configuration procedures and examples, refer to the chapter "Traffic Policies" in the *Ruckus FastIron Traffic Management Configuration Guide*.

**log**

Enables SNMP traps and Syslog messages for the rule. In addition, logging must be enabled using the **acl-logging** command.

**mirror**

Mirrors packets matching the rule.

# Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

# Usage Guidelines

Extended ACLs permit or deny traffic according to source and destination addresses, port protocol, and other IPv4 frame content. You can also enable logging and mirroring.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list. Such a rule is automatically assigned the next multiple of 10 as a sequence number.

You can specify a mask in either of the following ways:

- Wildcard mask format (for example, 0.0.0.255). The advantage of this format is that it enables you mask any bit, for example by specifying 0.255.0.255.
- Classless Interdomain Routing (CIDR) format, in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 in the wildcard mask format.

If you specify **icmp** and also specify the **any-icmp-type** option, the following QoS options are not available: **dscp-marking**, **dscp-matching**, **internal-priority-marking**, **802.1p-priority-marking**, ond **802.1p-priority-matching**.

On the Ruckus ICX 7150 and Ruckus ICX 7750, ACL logging is not supported for egress ACLs.

When specifying type of service (ToS), you can indicate multiple *tos-value* options by entering the sum of the needed ToS options. For example, to specify both **max-reliability** and **min-delay**, enter **10**. To specify all options, enter **15**. Values range from **0** through **15**.

In a rule that includes one or more of the following parameters, the **log** keyword is ignored:

- **dscp-matching**
- **dscp-marking**
- **802.1p-priority-matching**
- **802.1p-priority-marking**
- **802.1p-and-internal-marking**

For details on 802.1p priority matching, refer to "Inspecting the 802.1p bit in the ACL for adaptive rate limiting" in the *Ruckus FastIron Traffic Management Configuration Guide*.

To delete a rule from an ACL, do either of the following:

- Enter **no sequence** *seq-value*.
- Type **no** followed by the full command syntax without the **sequence** *seq-value*.

# Examples

The following ACL, applied to an Ethernet interface, blocks and logs IPv4 TCP packets transmitted by Telnet from a specified host to any destination.

```
device# configure terminal
device(config)# ip access-list extended "block Telnet"
device(config-ext-nacl)# sequence 10 deny tcp host 10.157.22.26 any eq telnet log
device(config-ext-nacl)# sequence 20 permit ip any any
device(config-ext-nacl)# exit
device(config)# interface ethernet 1/1/1
device(config-if-1/1/1)# ip access-group "block Telnet" in
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was modified to support the **sequence** keyword and to support logging in permit rules. |

# sequence (permit | deny in IPv6 ACLs)

Inserts filtering rules in IPv6 access control lists (ACLs).

## Syntax

Use the following syntax to define a TCP or UDP rule:

[ **sequence** *seq-num* ] { **deny** | **permit** } { **tcp** | **udp** } { *ipv6-source-prefix*/*prefix-length* | **host** *source-ipv6_address* | **any** } [ *source-comparison-operators* ] { *ipv6-destination-prefix*/*prefix-length* | **host** *ipv6-destination-address* | **any** } [ **established** ] [ *destination-comparison-operators* ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define an ICMP rule:

[ **sequence** *seq-num* ] { **deny** | **permit** } **icmp** { *ipv6-source-prefix*/*prefix-length* | **host** *source-ipv6_address* | **any** } { *ipv6-destination-prefix*/*prefix-length* | **host** *ipv6-destination-address* | **any** } [ *icmp-num* | *icmp-type* ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define an IPv6 rule:

[ **sequence** *seq-num* ] { **deny** | **permit** } **IPv6** { *ipv6-source-prefix*/*prefix-length* | **host** *source-ipv6_address* | **any** } { *ipv6-destination-prefix*/*prefix-length* | **host** *ipv6-destination-address* | **any** } [ **fragments** | **routing** ] [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

Use the following syntax to define an AHP, ESP, SCTP, *protocol-name*- or *protocol-number* rule:

[ **sequence** *seq-num* ] { **deny** | **permit** } { **AHP** | **ESP** | **SCTP** | *protocol-name* | *protocol-number* } { *ipv6-source-prefix*/*prefix-length* | **host** *source-ipv6_address* | **any** } { *ipv6-destination-prefix*/*prefix-length* | **host** *ipv6-destination-address* | **any** } [ **dscp-matching** *dscp-value* ] [ **dscp-marking** *dscp-value* ] [ **802.1p-priority-matching** *802.1p-value* ] [ **802.1p-priority-marking** *802.1p-value* ] [ **internal-priority-marking** *queuing-priority* ] [ **traffic-policy** *name* ] [ **log** ] [ **mirror** ]

**no sequence** *seq-num*

## Parameters

**sequence**

(Optional) Enables you to assign a sequence number to the rule.

*seq-num*

Valid values range from 1 through 65000.

**deny**

Specifies rules to deny traffic.

**permit**

Specifies rules to permit traffic.

*protocol-name* | *protocol-number*

Specifies the type of IPv6 packet you are filtering. You can specify one of the following protocol names or a valid protocol number (from 0 through 255).

- **ahp**: Authentication Header
- **esp**: Encapsulating Security Payload

- **icmp**: Internet Control Message Protocol
- **ipv6**: Internet Protocol, version 6
- **sctp**: Stream Control Transmission Protocol
- **tcp**: Transmission Control Protocol
- **udp**: User Datagram Protocol

*ipv6-source-prefix* **/** *prefix-length*

Specifies a source prefix and prefix length that a packet must match for the specified action (deny or permit) to occur. You must specify the *ipv6-source-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. Specify the *prefix-length* parameter as a decimal value, preceded by a slash mark (/).

**host** *source-ipv6_address*

Specifies a host source IPv6 address. When you use this parameter, you do not need to specify the prefix length. A prefix length of 128 is implied.

**any**

Specifies all source addresses.

*source-comparison-operators* and *destination-comparison-operators*

If you specified **tcp** or **udp**, the following optional operators are available:

**eq**

Specifies the port name or number you enter after **eq**.

**gt**

Specifies port numbers equal to or greater than the port number or equal to or greater than the numeric equivalent of the port name you enter after **gt**.

**lt**

Specifies port numbers that are less than or equal to the port number or less than or equal to the numeric equivalent of the port name you enter after **lt**.

**neq**

Specifies all port numbers except the port number or port name you enter after **neq**.

**range**

Specifies all port numbers that are between the first port name or number and the second one you enter following the **range** keyword. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: **range 23 53** (two values separated by a space). The first port number in the range must be lower than the last number in the range.

*ipv6-destination-prefix* **/** *prefix-length*

Specifies a destination prefix and prefix length that a packet must match for the specified action (deny or permit) to occur. You must specify the *ipv6-destination-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. Specify the *prefix-length* parameter as a decimal value, preceded by a slash mark (/).

**host** *destination-ipv6_address*

Specifies a destination host IPv6 address. When you use this parameter, you do not need to specify the prefix length. A prefix length of 128 is implied.

**any**

Specifies all destination addresses.

**established**

(For TCP only) Filter packets that have the Acknowledgment (ACK) or Reset (RST) flag set. This policy applies only to established TCP sessions, not to new sessions.

*icmp-num*

Specifies a numbered message type.

*icmp-type*

(For ICMP only) Specifies a named message type, from the following list.

**beyond-scope**

Specifies a beyond scope message.

**destination-unreachable**

Specifies a destination unreachable message.

**echo-reply**

Specifies an echo reply.

**echo-request**

Specifies an echo request (ping).

**header**

Specifies a parameter problem header error message.

**hop-limit**

Specifies an in-transit, time exceeded message.

**mld-query**

Specifies an MLD query message.

**mld-reduction**

Specifies an MLD reduction message.

**mld-report**

Specifies an MLD report message.

**nd-na**

Specifies a neighbor discovery (ND) neighbor advertisement message.

**nd-ns**

Specifies an ND neighbor solicitation message.

**next-header**

Specifies a parameter problem next-header error message.

**no-admin**

Specifies a destination unreachable administratively prohibited message.

**no-route**

Specifies a destination unreachable no route message.

**packet-too-big**

Specifies a packet too big message.

**parameter-option**

Specifies a parameter-option problem message.

**parameter-problem**

Specifies a parameter problem message.

**port-unreachable**
Specifies a destination-port unreachable message.

**reassembly-timeout**
Specifies a reassembly timeout message.

**renum-command**
Specifies a renumber command message.

**renum-result**
Specifies a renumber result message.

**renum-seq-number**
Specifies a renumber sequence number message.

**router-advertisement**
Specifies a router advertisment message.

**router-renumbering**
Specifies a router renumbering message.

**router-solicitation**
Specifies a router solicitation message.

**time-exceeded**
Specifies a time exceeded message.

**unreachable**
Specifies a destination-unreachable message.

**fragments**
(For IPv6 protocol only) Specifies fragmented packets that contain a non-zero offset.

**routing**
(For IPv6 protocol only) Specifies source-routed packets.

**dscp-matching** *dscp-value*
Filters by DSCP value. Values range from 0 through 63.

**dscp-marking** *dscp-value*
Assigns the DSCP value that you specify to the packet. Values range from 0 through 63.

**802.1p-priority-matching** *802.1p-value*
Filters by 802.1p priority, for rate limiting. Values range from 0 through 7.

**802.1p-priority-marking** *802.1p-value*
Assigns the 802.1p value that you specify to the packet. Values range from 0 through 7.

**internal-priority-marking** *queuing-priority*
Assigns the internal queuing priority (traffic class) that you specify to the packet. Values range from 0 through 7.

**traffic-policy** *name*
Enables the device to limit the rate of inbound traffic and to count the packets and bytes per packet to which ACL permit or deny clauses are applied.

**log**
Enables SNMP traps and syslog messages for the rule.

**mirror**
Mirrors packets matching the rule.

# Modes

ACL configuration mode

# Usage Guidelines

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list. Such a rule is automatically assigned the next multiple of 10 as a sequence number.

On the Ruckus ICX 7150 and Ruckus ICX 7750, ACL logging is not supported for egress ACLs.

In a rule that includes one or more of the following parameters, the **log** keyword is ignored:

- **dscp-matching**
- **dscp-marking**
- **802.1p-priority-matching**
- **802.1p-priority-marking**

To enable hop-limit check for the ACL, enter the **enable nd hop-limit** command from IPv6 ACL configuration mode.

For traffic policy configuration procedures and examples, refer to "Traffic Policies" in the *Ruckus FastIron Traffic Management Configuration Guide*.

To delete a rule from an ACL, do either of the following:

- Enter **no sequence** *seq-value*.
- Type **no** followed by the full command syntax without **sequence** *seq-value*.

For details on 802.1p rate limiting, refer to "Inspecting the 802.1p bit in the ACL for adaptive rate limiting" in the *Ruckus FastIron Traffic Management Configuration Guide*.

For the **log** keyword to trigger a log entry, logging must be enabled with the **logging-enable** command.

# Examples

The following example creates an IPv6 ACL named "netw", with remarks preceding each rule.

```
device# configure terminal
device(config)# ipv6 access-list netw

device(config-ipv6-access-list netw)# remark Permits ICMP traffic from 2001:DB8:e0bb::x to 2001:DB8::x.
device(config-ipv6-access-list netw)# sequence 10 permit icmp 2001:DB8:e0bb::/64 2001:DB8::/64

device(config-ipv6-access-list netw)# remark Denies traffic from 2001:DB8:e0ac::2 to 2001:DB8:e0aa:
0::24.
device(config-ipv6-access-list netw)# sequence 20 deny ipv6 host 2001:DB8:e0ac::2 host 2001:DB8:e0aa:
0::24

device(config-ipv6-access-list netw)# remark Denies all UDP traffic.
device(config-ipv6-access-list netw)# sequence 30 deny udp any any

device(config-ipv6-access-list netw)# remark Permits traffic not explicitly denied by the previous
rules.
device(config-ipv6-access-list netw)# sequence 40 permit ipv6 any any
```

The following example applies "netw" to incoming traffic on ports 1/1/2 and 1/4/3.

```
device# configure terminal
device(config)# interface ethernet 1/1/2
device(config-if-e1000-1/1/2)# ipv6 enable
device(config-if-e1000-1/1/2)# ipv6 traffic-filter netw in
device(config-if-e1000-1/1/2)# exit
device(config)# interface ethernet 1/4/3
device(config-if-e1000-1/4/3)# ipv6 enable
device(config-if-e1000-1/4/3)# ipv6 traffic-filter netw in
```

The following example creates an IPv6 ACL named "rtr", with remarks preceding each rule.

```
device# configure terminal
device(config)# ipv6 access-list rtr

device(config-ipv6-access-list rtr)# remark Denies TCP traffic from 2001:DB8:21::x to 2001:DB8:22::x.
device(config-ipv6-access-list rtr)# deny tcp 2001:DB8:21::/24 2001:DB8:22::/24

device(config-ipv6-access-list rtr)# remark Denies UDP traffic from UDP ports 5 through 6 to
2001:DB8:22::/24.
device(config-ipv6-access-list rtr)# deny udp any range 5 6 2001:DB8:22::/24

device(config-ipv6-access-list rtr)# remark Permits traffic not explicitly denied by the previous rules.
device(config-ipv6-access-list rtr)# permit ipv6 any any
```

The following example applies "rtr" to incoming traffic on ports 1/2/1 and 1/2/2.

```
device# configure terminal
device(config)# interface ethernet 1/2/1
device(config-if-e1000-1/2/1)# ipv6 enable
device(config-if-e1000-1/2/1)# ipv6 traffic-filter rtr in
device(config-if-e1000-1/2/1)# exit
device(config)# int eth 1/2/2
device(config-if-e1000-1/2/2)# ipv6 enable
device(config-if-e1000-1/2/2)# ipv6 traffic-filter rtr in
```

The following are examples of show command output for the ACL "rtr". Note that sequence numbers were automatically assigned.

```
device# show running-config
ipv6 access-list rtr
10: deny tcp 2001:DB8:21::/24 2001:DB8:22::/24
20: deny udp any range rje 6 2001:DB8:22::/24
30:permit ipv6 any anyy

device# show ipv6 access-list rtr
ipv6 access-list rtr: 3 entries
10: deny tcp 2001:DB8:21::/24 2001:DB8:22::/24
20: deny udp any range rje 6 2001:DB8:22::/24
30: permit ipv6 any any
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to support the **sequence** keyword and to support logging in permit rules. |

# sequence (permit | deny in standard IPv4 ACLs)

Inserts filtering rules in IPv4 standard named or numbered ACLs. Standard ACLs permit or deny traffic according to source address only.

## Syntax

**sequence** *seq-num* { **deny** | **permit** } { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

{ **deny** | **permit** } { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

**no sequence** *seq-num*

**no** { **deny** | **permit** } { *S_IPaddress* [ *mask* ] | **host** *S_IPaddress* | **any** } [ **log** ] [ **mirror** ]

## Parameters

**sequence**

(Optional) Enables you to assign a sequence number to the rule.

*seq-num*

Valid values range from 1 through 65000.

**deny**

Specifies rules to deny traffic.

**permit**

Specifies rules to permit traffic.

*S_IPaddress*

Specifies a source address for which you want to filter the subnet.

*mask*

Defines a subnet mask that includes the source address you specified.

**host**

Indicates the source IP address is a host address.

*S_IPaddress*

Specifies source address.

**any**

Specifies all source addresses.

**log**

Enables logging for the rule.

**mirror**

Mirrors packets matching the rule.

## Modes

IPv4 ACL configuration mode

IPv6 ACL configuration mode

# Usage Guidelines

This command configures rules to permit or drop traffic based on source addresses. You can also enable logging and mirroring.

The order of the rules in an ACL is critical, as the first matching rule stops further processing. When creating rules, specifying sequence values determines the order of rule processing. If you do not specify a sequence value, the rule is added to the end of the list. Such a rule is automatically assigned the next multiple of 10 as a sequence number.

You can specify a mask in either of the following ways:

- Wildcard mask format. The advantage of this format is that it enables you to mask any bit, for example by specifying 0.255.0.255.
- Classless Interdomain Routing (CIDR) format—in which you specify the number of bits of the prefix. For example, appending /24 to an IPv4 address is equivalent to specifying 0.0.0.255 in the wildcard mask format.

On Ruckus ICX 7150 and Ruckus ICX 7750 devices, ACL logging is not supported for egress ACLs.

For the **log** keyword to trigger a log entry, logging must be enabled with the **acl-logging** command.

To delete a rule from an ACL, do either of the following:

- Enter **no sequence** *seq-value*.
- Type **no** followed by the full command syntax without **sequence** *seq-value*.

# Examples

The following example shows how to configure a standard numbered ACL and apply it to incoming traffic on port 1/1/1.

```
device# configure terminal
device(config)# ip access-list standard 1
device(config-std-nacl)# sequence 10 deny host 10.157.22.26 log
device(config-std-nacl)# sequence 20 deny 10.157.29.12 log
device(config-std-nacl)# sequence 30 deny host IPHost1 log
device(config-std-nacl)# sequence 40 permit any
device(config-std-nacl)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# ip access-group 1 in
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to support the **sequence** keyword and to support logging in permit rules. |

# set ip next-hop

Configures the next-hop IP address for the traffic that matches a match statement in the route map.

## Syntax

**set ip next-hop** { **peer-address** | *ip-address* [ **no-ttl-decrement** | **vrf** *vrf-name* ] }

**no set ip next-hop** { **peer-address** | *ip-address* [ **no-ttl-decrement** | **vrf** *vrf-name* ] }

## Command Default

The next-hop IP address is not configured by default.

## Parameters

**peer-address**
Specifies the BGP peer IP address.

*ip-address*
Specifies the IP address of the next hop.

**no-ttl-decrement**
Disables the TTL value decrement and ensures that the packets are forwarded to the neighbor router without decrementing Time-to-Live (TTL) for the matched traffic.

**vrf** *vrf-name*
Specifies the VRF of the interface.

## Modes

Route map configuration mode

## Usage Guidelines

**no-ttl-decrement**

Policy-based routing (PBR) does not support the **peer-address** option while configuring the next-hop IP address using the **set ip next-hop** command.

For PBR on an interface in a VRF, if the VRF is not specified in the next hop (that is, only the IP address is specified as the next hop), the default VRF of the interface is considered. The next hop in a route map will take effect only if the interface on which the route map is applied and the next hop in the route map are in the same VRF

The **no-ttl-decrement** option is supported only on the Ruckus ICX 7750 and Ruckus ICX 7450 devices.

The **no** form of the command removes the next-hop IP address configured for the traffic.

# Examples

The following example configures a route map without decrementing the Time-to-Live (TTL) value.

```
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match ip address 100
device(config-routemap test-route)# set ip next-hop 192.168.3.1 no-ttl-decrement
device(config-routemap test-route)# exit
```

The following example configures a route map with the default VRF of the interface as the next hop.

```
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match ip address 100
device(config-routemap test-route)# set ip next-hop 192.168.3.1
device(config-routemap test-route)# exit
```

The following example configures a route map which specifies the next hop is a VRF named as vrf_c.

```
device(config)# route-map test-route permit 99
device(config-routemap test-route)# match ip address 100
device(config-routemap test-route)# set ip next-hop 192.168.3.1 vrf vrf_c
device(config-routemap test-route)# exit
```

# History

| Release version | Command history |
|---|---|
| 08.0.10d | The **no-ttl-decrement** option was introduced. |
| 08.0.30 | The support for the **no-ttl-decrement** option was added in 08.0.30 and later releases. |
| 08.0.50 | The **vrf** option was introduced. |

# send-lifetime

Configures the time period during which the key on a keychain becomes active and is valid to be sent.

## Syntax

**send-lifetime** [ **local** | **start** { *start-date start-time* **end** { *duration* | **infinite** | *end-date end-time* } } ]

**no send-lifetime**

## Command Default

The lifetime of send keys is not configured by default.

## Parameters

**local**
Specifies that the time zone used will be the time zone configured in the system.

**start**
Configures the point of time from which the key is valid to be sent.

*start-date*
Configures the start date in the *dd-mm-yy* format.

*start-time*
Configures the start time in the *hh:mm:ss* format.

**end**
Configures the point of time at which the send key expires.

*duration*
Configures the duration in seconds before the send key expires. The value ranges from 1 through 2147483646 seconds.

**infinite**
Configures the send key to never expire.

*end-date*
Configures the end date in the *dd-mm-yy* format.

*end-time*
Configures the end time in the *hh:mm:ss* format.

## Modes

Key ID configuration mode

## Usage Guidelines

All participating routers must have Network Time Protocol (NTP) enabled before setting the lifetime on the keys.

If the tolerance value is configured, the start time of send key to become active is advanced (start time minus tolerance) and the end time is moved further ahead (end time plus tolerance) before the key expires, unless the end time is set to be infinite.

A key is considered valid even when it is in the tolerance period.

A key can be selectively active for the accept lifetime and not the send lifetime.

The key must be configured with a minimum time of ten seconds.

The **no** form of the command negates the entire send lifetime and not merely individual options of the duration.

# Examples

The following example configures the time period during which the key on a keychain becomes active and valid to be sent.

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)# key-id 10
device(config-keychain-xprotocol-key-10)# send-lifetime start 10-10-17 10:10:10 end 10000
```

# History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# sflow agent-ip

Configures an arbitrary IPv4 or IPv6 address as the sFlow agent IP address.

## Syntax

**sflow agent-ip** { *ipv4-addr* | *ipv6-addr* }

**no sflow agent-ip** { *ipv4-addr* | *ipv6-addr* }

## Command Default

By default, the device automatically selects the sFlow agent IP address based on the configuration.

## Parameters

*ipv4-addr*

Specifies an IPv4 address as the sFlow agent IP address.

*ipv6-addr*

Specifies an IPv6 address as the sFlow agent IPv6 address.

## Modes

Global configuration mode

## Usage Guidelines

The sampled sFlow data that is sent to the collectors includes an agent_address field. This field identifies the device (the sFlow agent) that sent the data. By default, the device automatically selects the sFlow agent IP address based on the configuration. Alternatively, you can configure the device to use an arbitrary IPv4 or IPv6 address as the sFlow agent IP address instead.

The **no** form of the command removes the configured IPv4 or IPv6 address as the sFlow agent IP address.

## Examples

The following example configures an IPv4 address as the sFlow agent IP address.

```
device(config)# sflow agent-ip 10.10.10.1
```

The following example configures an IPv6 address as the sFlow agent IP address.

```
device(config)# sflow agent-ip FE80::240:D0FF:FE48:4672
```

# sflow destination

Configures an sFlow collector for the destination address.

## Syntax

**sflow destination** [ *ip-address* | **ipv6** *ipv6-address* ] [ *udp-port-number* ] [ **vrf** *vrf-name* ]

**no sflow destination** [ *ip-address* | **ipv6** *ipv6-address* ] [ *udp-port-number* ] [ **vrf** *vrf-name* ]

## Command Default

An sFlow collector is not configured.

## Parameters

*ip-address*
: Specifies the IPv4 destination address.

**ipv6** *ipv6-address*
: Specifies the IPv6 destination address.

*udp-port-number*
: Specifies the User Datagram Protocol (UDP) port number. The default value is 6343.

**vrf** *vrf-name*
: Specifies the Virtual Routing and Forwarding (VRF) name.

## Modes

Global configuration mode

## Usage Guidelines

sFlow exports traffic statistics to an external collector. You can specify up to four collectors. You can specify more than one collector with the same IP address if the UDP port numbers are unique. You can have up to four unique combinations of IP addresses and UDP port numbers.

By default sFlow uses the management VRF to send the samples to the collector. If no management VRF is configured, sFlow uses the default VRF, and this default VRF ID will be assigned to any configured collector that does not have a user-included VRF.

sFlow-forwarding ports can come from ports that belong to any VRF. The port does not have to be in the same VRF as the collector. sFlow collects packets from all sFlow-forwarding ports, even if they do not belong to a VRF, compiles the packets into the sFlow samples, and sends the samples to the particular collector with no filtering for VRF membership.

The **no** form of the command configures the management VRF to send the samples to the collector.

# Examples

The following example configures an sFlow collector and specifies a VRF.

```
device(config)# sflow destination 10.10.10.10 vrf customer1
```

# sflow enable

Enables sFlow forwarding globally.

## Syntax

**sflow enable**

**no sflow enable**

## Command Default

sFlow is not enabled.

## Modes

Global configuration mode

## Usage Guidelines

To enable sFlow forwarding, you must first enable it on a global basis and then use the **sflow forwarding** command to enable it on individual interfaces or LAG ports, or both.

The **no** form of the command disables sFlow forwarding globally.

## Examples

The following example enables sFlow forwarding globally.

```
device(config)# sflow enable
```

# sflow export

Configures exporting, to the sFlow collector, the CPU usage and memory usage information or exporting CPU-directed data.

## Syntax

**sflow export** { **cpu-traffic** [ *traffic-seconds* ] | **system-info** [ *info-seconds* ] }

**no sflow export** { **cpu-traffic** [ *traffic-seconds* ] | **system-info** [ *info-seconds* ] }

## Command Default

By default, CPU and memory usage information and CPU-directed data are not exported.

## Parameters

**cpu-traffic**
Specifies the CPU usage.

*traffic-seconds*
Specifies the average sampling rate of incoming packets on an sFlow-enabled port to the number of flow samples taken from those packets.

**system-info**
Specifies the system information and the memory usage.

*info-seconds*
Specifies the polling interval, in seconds. The default polling interval for exporting CPU and memory usage information to the sFlow collector is 20 seconds and the interval for exporting CPU-directed data to the sFlow collector is 16.

## Modes

Global configuration mode

## Usage Guidelines

The polling interval defines how often sFlow data for a port is sent to the sFlow collector.

The **no** form of the command removes the configured value and sets the sampling rate or the polling interval to its default value.

## Examples

The following example sets the sampling rate to 2048.

```
device(config)# sflow export cpu-traffic 2048
```

The following example enables the export of CPU usage and memory usage information.

```
device(config)# sflow export system-info
```

The following example sets the polling interval for exporting CPU and memory usage information to 30 seconds.

```
device(config)# sflow export system-info 30
```

# sflow forwarding

Enables sFlow forwarding on individual interfaces.

## Syntax

**sflow forwarding**

**no sflow forwarding**

## Command Default

sFlow forwarding is not enabled on individual interfaces.

## Modes

Interface configuration mode

## Usage Guidelines

You must use both the **sflow enable** command and the **sflow forwarding** command to enable the feature.

The **no** form of the command disables sFlow forwarding on individual interfaces.

## Examples

The following example enables sFlow forwarding on a range of Ethernet interfaces.

```
device(config)# sflow enable
device(config)# interface ethernet 1/1/1 to 1/1/8
device(config-mif-1/1/1-1/1/8)# sflow forwarding
```

# sflow forwarding (LAG)

Enables sFlow forwarding on an individual port in a deployed LAG.

## Syntax

**sflow forwarding** { **ethernet** *stackid/slot/port* | **port-name** *name* }

**no sflow forwarding** { **ethernet** *stackid/slot/port* | **port-name** *name* }

## Command Default

sFlow is not configured.

## Parameters

**ethernet** *stackid/slot/port*

Specifies the Ethernet port within the LAG on which you want to enable sFlow forwarding.

**port-name** *name*

Specifies a named port within the LAG on which you want to enable sFlow forwarding.

## Modes

LAG configuration mode

## Usage Guidelines

For a keep-alive LAG, sFlow can be enabled only in interface configuration mode and not in LAG configuration mode.

The **no** form of the command disables sFlow forwarding.

## Examples

The following example shows how to enable sFlow forwarding on an individual port.

```
device(config)# lag blue static id 1
device(config-lag-blue)# sflow forwarding ethernet 1/3/1
```

The following example shows how to enable sFlow forwarding on a named port.

```
device(config)# lag test2 static id 2
device(config-lag-test2)# sflow forwarding port-name port1
```

# sflow management-vrf-disable

Disables the management Vitual Routing and Forwarding (VRF) in sFlow.

## Syntax

**sflow management-vrf-disable**

**no sflow management-vrf-disable**

## Command Default

sFlow uses the management VRF to send samples to the collector.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command enables the management VRF in sFlow.

## Examples

The following example disables management VRF in sFlow.

```
device(config)# sflow management-vrf-disable
```

# sflow max-packet-size

Configures the maximum flow sample size sent to the sFlow collector.

## Syntax

**sflow max-packet-size** *size*

**no sflow max-packet-size**

## Command Default

The default maximum flow sample size is 128 bytes.

## Parameters

*size*

Specifies the maximum sFlow packet size, in bytes. For sFlow version 5, the maximum flow sample size is 1300 bytes.

## Modes

Global configuration mode

## Usage Guidelines

With sFlow version 5, you can specify the maximum size of the flow samples sent to the sFlow collector. If a packet is larger than the specified maximum size, only the contents of the packet up to the specified maximum number of bytes are exported. If the size of the packet is smaller than the specified maximum, the entire packet is exported.

The **no** form of the command removes the configured value and reverts to the default value.

## Examples

The following example sets the maximum flow sample size to 1024.

```
device(config)# sflow max-packet-size 1024
```

# sflow polling-interval

Configures the sflow polling interval.

## Syntax

**sflow polling-interval** *secs*

**no sflow polling-interval**

## Command Default

The default polling interval is 20 seconds.

## Parameters

*secs*

Specifies the polling interval, in seconds. The value can range from 0 through 429496729. If you specify 0, counter data sampling is disabled. The default polling interval is 20 seconds.

## Modes

Global configuration mode

## Usage Guidelines

The polling interval defines how often sFlow byte and packet counter data for a port is sent to the sFlow collectors. If multiple ports are enabled for sFlow, the device staggers transmission of the counter data to smooth performance. For example, if sFlow is enabled on two ports and the polling interval is 20 seconds, the device sends counter data every 10 seconds. The counter data for one of the ports is sent after 10 seconds, and the counter data for the other port is sent after an additional 10 seconds. 10 seconds later, new counter data for the first port is sent.

The interval value applies to all interfaces on which sFlow is enabled.

The **no** form of the command returns the polling interval to the default value.

## Examples

The following example sets the polling interval to 30 seconds.

```
device(config)# sflow polling-interval 30
```

# sflow sample

Changes the default sampling rate.

## Syntax

**sflow sample** *num*

**no sflow sample** *num*

## Command Default

The default sampling rate is 4096 packets.

## Parameters

*num*

Specifies the average number of packets from which each sample is taken. The software rounds the value that you enter to the next higher odd power of 2. Refer to the Usage Guidelines section for information on the range of supported values.

## Modes

Global configuration mode

Interface configuration mode

LAG configuration mode

## Usage Guidelines

The value range for the sampling rate on Ruckus ICX 7250, ICX 7450, and ICX 7750 is from 256 through 1073741823.

You cannot change a module's sampling rate directly. You can change a module's sampling rate only by changing the sampling rate of a port on that module.

You can configure an individual port to use a different sampling rate than the global default sampling rate. This is useful when ports have different bandwidths.

You can configure individual LAG ports to use a different sampling rate than the global default sampling rate. For a keep-alive LAG, sFlow can be enabled only at the interface level and not at the LAG level.

When configuring the sample rate, if you configure the value as 1000, the software rounds the value to the next higher odd power of 2; so the actual rate is 2^11 (2048), and 1 in 2048 packets are sampled by the hardware.

The **no** form of the command resets the sampling rate to the default value.

## Examples

The following example changes the default (global) sampling rate.

```
device(config)# sflow sample 2048
```

The following example changes the sampling rate on an individual port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# sflow sample 8192
```

The following example enables an sFlow sample rate in a LAG configuration.

```
device(config)# lag blue static id 1
device(config-lag-blue)# sflow sample 512
```

# sflow source

Configures the sFlow source interface (IPv4 or IPv6) from which the IP source address is selected for the sFlow datagram.

## Syntax

**sflow source** [ **ipv6** ] { **ethernet** *stackid/slot/port* | **ve** *ve-number* | **loopback** *number* }

**no sflow source** [ **ipv6** ] { **ethernet** *stackid/slot/port* | **ve** *ve-number* | **loopback** *number* }

## Command Default

The sFlow source is not configured. The IP address of the outgoing interface is used in the sFlow datagram.

## Parameters

**ipv6**

Configures the IPv6 interface as the sFlow source. If **ipv6** is not specified, the IPv4 interface is automatically configured as the sFlow source.

**ethernet** *stackid/slot/port*

Configures an Ethernet interface as the sFlow source interface.

**ve** *ve-number*

Configures a virtual interface (VE) as the sFlow source interface.

**loopback** *number*

Configures a loopback interface as the sFlow source interface.

## Modes

Global configuration mode

## Usage Guidelines

At any time, only one source of the Ethernet, VE, or loopback interface can be specified as the source interface.

The first IP address in the interface IP address list is considered the source IP address. Upon configuring another source for an IPv4 or IPv6 address, any previously configured source for the IPv4 or IPv6 address will be deleted. You can configure IPv4 and IPv6 source interfaces independently.

If the sFlow destination is IPv6, and the sFlow source is configured for an IPv6 address, then an IPv6 address will be selected from the configured interface. If the sFlow destination is IPv4, and the sFlow source is configured for an IPv4 address, then an IPv4 address will be selected from the configured interface.

The **no** form of the command removes the sFlow source configuration from the interface and restores the default behavior of using IP address of the outgoing interface as the source IP address of the sFlow datagram.

# Examples

The following example configures an Ethernet interface to be used as the sFlow source IPv6 interface.

```
device(config)# sflow source ipv6 ethernet 1/1/2
```

The following example configures an Ethernet interface to be used as the sFlow source IPv4 interface.

```
device(config)# sflow source ethernet 1/1/3
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# sflow source-port

Configures the source sFlow UDP port.

## Syntax

**sflow source-port** *num*

**no sflow source-port**

## Command Default

sFlow sends data to the collector using UDP source port 8888.

## Parameters

*num*

Specifies the sFlow source port. The value can range from 1025 through 65535.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command reverts the sFlow source port to its default port of 8888.

## Examples

The following example changes the source UDP port to 8000.

```
device(config)# sflow source-port 8000
```

# sflow version

Configures the version used for exporting sFlow data.

## Syntax

**sflow version** *version-num*

**no sflow version** [ *version-num* ]

## Command Default

When sFlow is enabled globally on the device, the sFlow agent exports sFlow data in version 5 format.

## Parameters

*version-num*

Specifies the version number. The version can be 2 or 5.

## Modes

Global configuration mode

## Usage Guidelines

You can switch between versions without rebooting the device or disabling sFlow.

**NOTE**
When the sFlow version number is changed, the system resets sFlow counters and flow sample sequence numbers.

The **no** form of the command resets the sFlow verison to its default.

## Examples

The following example sets the sFlow version to 2.

```
device(config)# sflow version 2
```

# short-path-forwarding

Enables short-path forwarding on a Virtual Router Redundancy Protocol (VRRP) router.

## Syntax

**short-path-forwarding** [ **revert-priority** *number* ]

**no short-path-forwarding** [ **revert-priority** *number* ]

## Command Default

Short-path forwarding is disabled.

## Parameters

**revert-priority** *number*

Allows additional control over short-path forwarding on a backup router. If you configure this option, the revert-priority number acts as a threshold for the current priority of the session, and only if the current priority is higher than the revert-priority will the backup router be able to route frames. The range of revert-priority is 1 to 254.

## Modes

VRRP-E router configuration mode

## Usage Guidelines

Short-path forwarding means that a backup physical router in a virtual router attempts to bypass the VRRP-E master router and directly forward packets through interfaces on the backup router.

This command can be used for VRRP-E, but not for VRRP. You can perform this configuration on a virtual Ethernet (VE) interface only.

Enter the **no short-path-forwarding** command to remove this configuration.

## Examples

To enable short-path forwarding for a VRRP-E instance:

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# slow-start 40
device(config-vrrpe-router)# short-path-forwarding
```

# site (vxlan)

Create one remote site for the VXLAN overlay-gateway.

## Syntax

**site** *site-name*

**no site** *site-name*

## Command Default

Remote site is not configured.

## Parameters

*site-name*

Specifies the name of the remote site.

## Modes

Overlay-gateway configuration mode

## Usage Guidelines

The **no** form of the command removes the configured remote site.

A maximum of 32 remote sites (VXLAN tunnels) can be configured.

The command is supported only on ICX 7750 devices.

## Examples

The following example configures remote site site1.

```
device# configure terminal
device(config)#overlay-gateway gate1
device(config-overlay-gw-gate1)# site site1
device(config-overlay-gw-gate1-site1)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# Show Commands

## show 802-1w

Displays the Rapid Spanning Tree Protocol (RSTP) information of the specified port-based VLAN.

### Syntax

**show 802-1w** [ **detail** ] [ *number* | **vlan** *vlan-id* ]

### Parameters

**detail**

Displays detailed output.

*number*

Specifies the number of spanning tree entries to skip before the display begins.

**vlan** *vlan-id*

Displays the RSTP details for a specific VLAN.

### Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Cluster configuration mode

VLAN configuration mode

# Examples

The following example shows the output of the **show 802-1w** command.

```
device# show 802-1w
--- VLAN 4 [ STP Instance owned by VLAN 4 ] ----------------------------
Bridge IEEE 802.1W Parameters:
Bridge             Bridge     Bridge    Bridge     Force       tx
Identifier         MaxAge     Hello     FwdDly     Version     Hold
hex                sec        sec       sec                                                    cnt
8000002022227700   20         2         15         Default    3
RootBridge         RootPath      DesignatedBri-    Root    Max    Fwd   Hel
Identifier         Cost          dge Identifier    Port    Age    Dly   lo
hex
hex                                              sec    sec   sec
8000002022227700   0             8000002022227700  Root    20     15    2
Port IEEE 802.1W Parameters:
<--- Config Params --><-------------- Current state ----------------->
Port    Pri    PortPath    P2P     Edge      Role       State       Designa-     Designated
Num            Cost        Mac     Port                             ted cost     bridge
1/1/1   128    20000       F       F         DESIGNATED FORWARDING  0            8000002022227700
--- VLAN 5 [ STP Instance owned by VLAN 5 ] ----------------------------
Bridge IEEE 802.1W Parameters:
Bridge           Bridge     Bridge       Bridge     Force       tx
Identifier       MaxAge     Hello        FwdDly     Version     Hold
hex              sec        sec          sec                    cnt
8000002022227700 20         2            15         Default     3
RootBridge       RootPath       DesignatedBri-      Root    Max   Fwd   Hel
Identifier       Cost           dge Identifier      Port    Age   Dly   lo
hex                             hex                        sec   sec   sec
8000002022227700 0             8000002022227700    Root    20    15    2
Port IEEE 802.1W Parameters:
<--- Config Params --><-------------- Current state ----------------->
Port     Pri   PortPath    P2P    Edge       Role        State       Designa-      Designated
Num            Cost        Mac    Port                               ted cost      bridge
1/1/1    128   20000       F      F          DESIGNATED  FORWARDING  0             8000002022227700
--- VLAN 6 [ STP Instance owned by VLAN 6 ] ----------------------------
Bridge IEEE 802.1W Parameters:
Bridge           Bridge     Bridge        Bridge     Force        tx
Identifier       MaxAge     Hello         FwdDly     Version      Hold
hex              sec        sec           sec                     cnt
8000002022227700 20         2             15         Default      3
RootBridge       RootPath     DesignatedBri-     Root       Max   Fwd   Hel
Identifier       Cost         dge Identifier     Port       Age   Dly   lo
hex                           hex                           sec   sec   sec
8000002022227700 0            8000002022227700   Root       20    15    2
Port IEEE 802.1W Parameters:
<--- Config Params --><-------------- Current state ----------------->
Port   Pri     PortPath    P2P     Edge       Role        State       Designa-       Designated
Num            Cost        Mac     Port                               ted cost       bridge
1/1/1  128     20000       F       F          DESIGNATED  FORWARDING  0              8000002022227700
```

# show aaa

Displays information about all TACACS+ and RADIUS servers identified on the device.

## Syntax

**show aaa**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show aaa** command displays the following information:

| Output field | Description |
|---|---|
| Tacacs+ key | The setting configured with the **tacacs-server key** command. At the Super User privilege level, the actual text of the key is displayed. At the other privilege levels, a string of periods (....) is displayed instead of the text. |
| Tacacs+ retries | The setting configured with the **tacacs-server retransmit** command. |
| Tacacs+ timeout | The setting configured with the **tacacs-server timeout** command. |
| Tacacs+ dead-time | The setting configured with the **tacacs-server dead-time** command. |
| Tacacs+ Server | For each TACACS/TACACS+ server, the IP address, port, and the following statistics are displayed:<br>• opens - Number of times the port was opened for communication with the server<br>• closes - Number of times the port was closed normally<br>• timeouts - Number of times the port was closed due to a timeout<br>• errors - Number of times an error occurred while opening the port<br>• packets in - Number of packets received from the server<br>• packets out - Number of packets sent to the server |
| connection | The current connection status. This can be "no connection" or "connection active". |
| Radius key | The setting configured with the **radius-server key** command. At the Super User privilege level, the actual text of the key is displayed. At the other privilege levels, a string of periods (....) is displayed instead of the text. |
| Radius retries | The setting configured with the **radius-server retransmit** command. |
| Radius timeout | The setting configured with the **radius-server timeout** command. |
| Radius Server | For each RADIUS server, the IP address, and the following statistics are displayed:<br>• Auth Port - RADIUS authentication port number (default 1645)<br>• Acct Port - RADIUS accounting port number (default 1646)<br>• opens - Number of times the port was opened for communication with the server<br>• closes - Number of times the port was closed normally<br>• timeouts - Number of times the port was closed due to a timeout |

| Output field | Description |
|---|---|
| | • errors - Number of times an error occurred while opening the port<br>• packets in - Number of packets received from the server<br>• packets out - Number of packets sent to the server |
| connection | The current connection status. This can be "no connection" or "connection active". |

# Examples

The following example displays information about all TACACS/TACACS+ and RADIUS servers identified on the device.

```
device(config)# show aaa
Tacacs+ key: foundry
Tacacs+ retries: 1
Tacacs+ timeout: 15 seconds
Tacacs+ Server: 10.95.6.90 Port:49:
            opens=6 closes=3 timeouts=3 errors=0
            packets in=4 packets out=4
no connection
Radius key: networks
Radius retries: 3
Radius timeout: 3 seconds
Radius Server: 10.95.6.90 Auth Port=1645 Acct Port=1646:
            opens=2 closes=1 timeouts=1 errors=0
            packets in=1 packets out=4
no connection
```

# show access-list

Displays access control list (ACL) status information for a specific numbered ACL or for all named and numbered ACLs.

## Syntax

**show access-list** { *std-acl-num* | *extd-acl-num* | **all** | **hw-usage** { **on** | **off** } }

## Parameters

*std-acl-num*
: Displays information about the specified standard ACL. Valid values are from 1 through 99.

*extd-acl-num*
: Displays information about the specified extended ACL. Valid values are from 100 through 199.

**all**
: Displays information about all ACLs.

**hw-usage**
: Displays the hardware usage statistics.

**on**
: Enables display of ACL rule numbers needed by hardware.

**off**
: Disables display of ACL rule numbers needed by hardware.

## Modes

User EXEC mode

## Usage Guidelines

The number of configured ACL rules can affect the rate at which hardware resources are used. You can use the **show access-list hw-usage on** command to enable hardware usage statistics, followed by the **show access-list** *std-acl-num* command to determine the hardware usage for an ACL. To gain more hardware resources, you can modify the ACL rule so that it uses fewer hardware resources.

From FastIron release 08.0.50, sequence numbers are automatically added to existing ACL rules in the following manner:

- The first rule within each ACL is numbered 10.
- The sequence number for each succeeding rule is incremented by 10.

In FastIron release 08.0.40, an ACL name can no longer be entered directly as a command option. Refer to the command **show access-list named-acl**, which was introduced as a replacement.

## Command Output

The **show access-list all** command displays the following information:

| Output field | Description |
|---|---|
| Rule cam | Lists the number of CAM entries used by the ACL or entry. The number of CAM entries listed for the ACL itself is the total of the CAM entries used by the ACL entries. |
| Flows | Lists the number of Layer 4 session table flows in use for the ACL. |
| Packets | Lists the number of packets and is applicable only to flow-based ACLs. |

# Examples

The following example shows sample output from the **show access-list all** command.

```
device# show access-list all
Extended IP access list 100 (Total flows: N/A, Total packets: N/A, Total rule cam use: 3)
10: permit udp host 192.168.2.169 any (Flows: N/A, Packets: N/A, Rule cam use: 1)
20: permit icmp any any (Flows: N/A, Packets: N/A, Rule cam use: 1)
30: deny ip any any (Flows: N/A, Packets: N/A, Rule cam use: 1)
```

The following example shows sample output from the **show access-list all** command that includes hardware information.

```
device# show access-list all
Standard IP access list 1 (hw usage (if applied on 24GC modules) : 2) (hw usage (if applied on 48GC
modules) : 2)
10: permit any (hw usage (if applied on 24GC modules) : 1) (hw usage (if applied on 48GC modules) : 1)

Extended IP access list 100 (hw usage (if applied on 24GC modules) : 7) (hw usage (if applied on 48GC
modules) : 7)
10: deny tcp any range newacct src any (hw usage (if applied on 24GC modules) : 6) (hw usage (if
applied on 48GC modules) : 6)
```

The following example adds hardware usage statistics to the output of the **show access-list** command.

```
device# show access-list hw-usage on
device# show access-list 100
Extended IP access list 100 (hw usage : 2)
10: deny ip any any (hw usage : 1
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | The command was modified. ACL names are no longer supported as an optional argument. |
| 08.0.50 | The command was modified so that sequence numbers are automatically added to existing rules. |

# show access-list accounting

Displays the access control list (ACL) accounting statistics for IPv4 ACLs, IPv6 ACLs, and Layer 2 MAC filters.

## Syntax

**show access-list accounting** *interface-type interface-name* [ **in** | **out** ] [ **IPv4** | **IPv6** | **Mac** ] [ **detail** ]

**show access-list accounting traffic-policy** [ **all** | *name* ]

## Parameters

*interface-type interface-name*
Specifies the interface type (Ethernet, virtual interface, or LAG) and the ID of the interface.

**in**
Displays the statistics of the inbound ACLs. If no direction is set, statistics for both inbound and outbound are shown.

**out**
Displays the statistics of the outbound ACLs. If no direction is set, statistics for both inbound and outbound are shown.

**IPv4**
Displays the statistics for IPv4 ACLs. Statistics for both IPv4 and IPv6 ACLs are shown if this value is not set.

**IPv6**
Displays the statistics for IPv6 ACLs. Statistics for both IPv4 and IPv6 ACLs are shown if this value is not set.

**Mac**
Displays the statistics for Layer 2 MAC filters. This is valid for the inbound direction only on physical ports and LAG virtual interface ports.

**traffic-policy**
Displays traffic policy statistics.

**all**
Displays the statistics of all traffic policies.

*name*
Displays the statistics of a specific traffic policy.

*detail*
Displays a detailed report.

## Modes

Privileged EXEC mode

# Usage Guidelines

The output displayed gives information about IPv4 ACLs or IPv6 ACLs, or MAC filters based on the configuration of the port or interface. If both IPv4 and IPv6 ACLs are configured on the same port, the output provides both IPv4 and IPv6 ACL accounting information.

Note that all interfaces are specified using the parameters *interface-type* and *interface-id*. For example, to specify a LAG interface with an ID of 200, enter *lag 200*.

# Command Output

The **show access-list accounting** command displays the following information.

| Output field | Description |
|---|---|
| IPV4 ACL Accounting Information or IPv6 ACL Accounting Information | Denotes the ACL for which the accounting information was collected. |
| #: <ACL filter rule> | Shows the ACL filter sequence number followed by the ACL filter rule. For example: **10: permit ip host 1.1.1.9 any** |
| Port region: <port number> | Shown in detailed reports only. Denotes the first port on the device on which the ACL statistics are collected. |
| Total | Shows the aggregate statistics for the port. |
| Hit count | Shows the number of hits for each counter. |

# Examples

The following output shows a virtual interface that has both IPv4 and IPv6 ACLs applied to the same port and has ACL accounting enabled. This example includes the **detail** parameter to provide a detailed report.

```
device# show access-list accounting ve 16 out detail
IPV4 ACL Accounting Information for Outbound ACL: 101

10: permit ip host 1.1.1.9 any
    Port Region: 1/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Port Region: 2/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Port Region: 3/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Total:
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------
11: permit icmp any any
    Port Region: 1/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Port Region: 2/1/1
   Hit Count:   (1Min)                         0   (5Sec)                         0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Port Region: 3/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Total:
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------
15: permit ip host 10.1.1.15 any
    Port Region: 1/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Port Region: 2/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Port Region: 3/1/1
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------

    Total:
    Hit Count:   (1Min)                          0   (5Sec)                          0
                 (PktCnt)                        0 (ByteCnt)                        0
    ------------------------------------------------------------------------------
```

```
IPV4 ACL Accounting Information
devNum[0] => ACL: 10
    0: permit any
     Hit Count:    (1Min)            0    (5Sec)                          0
                   (PktCnt)          0  (ByteCnt)                         0
    ----------------------------------------------------------------------
65535: Implicit Rule deny any any
     Hit Count:    (1Min)            0    (5Sec)                          0
                   (PktCnt)          0  (ByteCnt)                         0
    ----------------------------------------------------------------------

IPV6 ACL Accounting Information
devNum[0] => ACL: v6
    0:  permit ipv6 any any
     Hit Count:    (1Min)             0    (5Sec)                         0
                   (PktCnt)           0   (ByteCnt)                       0
    ----------------------------------------------------------------------
65533: Implicit ND_NA Rule: permit any any
     Hit Count:    (1Min)              0    (5Sec)                        0
                   (PktCnt)            0  (ByteCnt)                       0
    ----------------------------------------------------------------------
65534: Implicit ND_NS Rule: permit any any
     Hit Count:    (1Min)              0    (5Sec)                        0
                   (PktCnt)            0  (ByteCnt)                       0
    ----------------------------------------------------------------------
65535: Implicit Rule: deny any any
     Hit Count:    (1Min)              0    (5Sec)                        0
                   (PktCnt)            0  (ByteCnt)                       0
    ----------------------------------------------------------------------
```

Note that in releases prior to FastIron 08.0.70, the **show access-list accounting** command shows a detailed report similar to the previous example. It is not possible to view an aggregate report in releases prior to FastIron 08.0.70.

The following output shows an aggregate report (rather than a detailed report) for a virtual interface with IPv4 outbound ACLs applied to the port.

```
device# show access account-list accounting ve 100 out
IPV4 ACL Accounting Information for Outbound ACL: 101
    10: permit ip any any
    Hit Count:    (1Min)                    4617559    (5Sec)                    421238
                  (PktCnt)                  4617559  (ByteCnt)                591048064
    ----------------------------------------------------------------------------------
65535: Implicit Rule deny any any
    Hit Count:    (1Min)                           0    (5Sec)                         0
                  (PktCnt)                         0  (ByteCnt)                        0
    ----------------------------------------------------------------------------------
```

The following output shows an Ethernet interface that has a MAC filter applied and ACL accounting enabled.

```
device# show access-list accounting ethernet 3/1/2 in

MAC Filters Accounting Information
    0: DA ANY SA 0000.0000.0001 - MASK FFFF.FFFF.FFFF
       action to take : DENY
     Hit Count:    (1Min)                       0    (5Sec)                  0
                   (PktCnt)                     0  (ByteCnt)                 0
    ------------------------------------------------------------------
65535: Implicit Rule deny any any
     Hit Count:    (1Min)            5028    (5Sec)              2129
                   (PktCnt)          5028  (ByteCnt)            643584
    ------------------------------------------------------------------
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |
| 08.0.70 | This command was modified to support outbound ACL accounting for IPv4 and IPv6 ACLs. |

# show access-list named-acl

Displays access control list (ACL) status information for a specific named ACL.

## Syntax

**show access-list named-acl** *acl-name*

## Parameters

*acl-name*

Specifies the named ACL.

## Modes

User EXEC mode

## Usage Guidelines

From FastIron release 08.0.50, sequence numbers are automatically added to existing ACL rules, in the following manner:

- The first rule within each ACL is numbered 10.
- The sequence number for each succeeding rule is incremented by 10.

## Examples

The following example displays information about "acl_01".

```
device# show access-list named-acl acl_01

Standard IP access list  acl_01 : 4 entries
10: deny host 10.157.22.26
20: deny 10.157.29.12
30: deny host IPHost1
40: permit any
```

## History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.50 | The command was modified to add sequence numbers automatically to existing rules. |

# show acl-on-arp

Displays the list of ACLs that are configured to filter ARP requests.

## Syntax

**show acl-on-arp** [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ] ] | **loopback** *num* | **tunnel** *num* | **ve** *num* ]

**show acl-on-arp** [ **lag** { **id** *lag-id* | **name** *lag-name* | *lag-id* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... }]

## Parameters

**ethernet** *unit/slot/port*
Displays the list of ACLs that are configured to filter ARP requests on a specific Ethernet interface.

**to** *unit/slot/port*
Displays the list of ACLs that are configured to filter ARP requests on a range of Ethernet interfaces.

**loopback** *num*
Displays the list of ACLs that are configured to filter ARP requests on a specific loopback interface.

**tunnel** *num*
Displays the list of ACLs that are configured to filter ARP requests on a specific tunnel interface.

**ve** *num*
Displays the list of ACLs that are configured to filter ARP requests on a specific VE interface.

**lag**
Displays the status of the LAG.

**id** *lag-id*
Displays the list by LAG ID.

**name** *lag-name*
Displays the list by LAG name.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

Access list configuration mode

## Usage Guidelines

The Filter Count column shows how many ARP packets have been dropped on the interface since the last time the count was cleared.

# Examples

The following example displays a sample output of the **show acl-on-arp** command.

```
device(config)# show acl-on-arp
Port    ACL ID        Filter Count
2        103          10
3        102            23
4        101            12
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# show arp

Displays the ARP table.

## Syntax

**show arp** [ *ip-addr* [ *ip-mask* ] | *num-entries-to-skip* | **ethernet** *unit/slot/port* | **mac-address** *xxxx.xxxx.xxxx* [ *xxxx.xxxx.xxxx* ] | **management** *port_number* | **resource** ] [ **|** *output_modifiers expression_string* ]

**show arp vrf** *vrf-name* [ *ip-addr* [ *ip-mask* ] | **ethernet** *unit/slot/port* | **mac-address** *xxxx.xxxx.xxxx* [ *xxxx.xxxx.xxxx* ] ] [ **|** *output_modifiers expression_string* ]

## Parameters

*ip_addr*
Specifies IP address.

*ip_mask*
Specifies IP subnet.

*num-entries-to-skip*
Number of entries to skip.

**ethernet** *unit/slot/port*
Displays specified ethernet port.

**mac-address** *xxxx.xxxx.xxxx* [ *xxxx.xxxx.xxxx* ]
Limits the output to the ARP entry that contains the specified MAC address. You may enter a second MAC address without re-entering the **mac-address** keyword to display information for two ARP entries.

**management** *port_number*
Limits the output to a specified management port.

**resource**
Limits the output to resource information.

**|** *output_modifiers expression_string*
Output modifiers that can follow the **|** symbol are **begin**, **include**, and **exclude**, which in turn are followed by an expression string that must be matched to restrict show command output.

**vrf** *vrf_name*
Displays ARP entries belonging to a given VRF instance.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

# Usage Guidelines

Use this command to view the total number of ARP entries and the maximum capacity for the ARP table along with the details of the ARP entries.

Use output modifiers to focus output if you wish.

# Command Output

The **show arp** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IP address of the entry. |
| MAC Address | The MAC address of the entry. |
| Type | ARP entry type. The options are :<br>• Static: The Layer 3 switch loaded the entry from the static ARP table when the device for the entry was connected to the Layer 3 switch.<br>• Dynamic: The Layer 3 switch learned the entry from an incoming packet.<br>• DHCP - The Layer 3 Switch learned the entry from the DHCP binding address table. In this case, the port number is not available until the entry gets resolved through ARP. |
| Age | The number of minutes since the ARP entry was refreshed. If this value reaches the defined ARP aging period, the entry is removed from the table. Static entries do not age out. |
| Port | Port associated with the ARP entry. |
| Status | Status of the ARP entry, either valid or pending. |

# Examples

The following example displays the ARP table.

```
device# show arp
Total number of ARP entries: 3
Entries in default routing instance:
No.   IP Address       MAC Address    Type     Age Port              Status
1     10.25.224.1      02e0.526a.3e3e Dynamic  0    mgmt1            Valid
2     10.25.224.2      001b.ed0c.c200 Dynamic  0    mgmt1            Valid
3     10.25.224.3      001b.ed0f.bc00 Dynamic  0    mgmt1            Valid
```

# show authentication acls

Displays information about currently active user-defined and dynamically applied ACLs.

## Syntax

**show authentication acls** { **all** | **ethernet** { *unit* **/** *slot* **/** *port* } | **unit** *unit_number* }

## Parameters

**all**

Shows ACLs for all stack units or for a standalone unit.

**ethernet** { *unit* **/** *slot* **/** *port* }

Shows ACLs for an interface or range of interfaces.

**unit** *unit_number*

Shows ACLs for the stack unit specified.

## Modes

Privileged EXEC mode or any configuration mode.

## Examples

The following example displays information on ACLs for all interfaces in the standalone unit or in the stack.

```
device# show authentication acls all
-------------------------------------------------------------------------------
Port MAC Address     V4 Ingress V4 Egress V6 Ingress V6 Egress
-------------------------------------------------------------------------------
1/1/7 0180.c200.0003 -          -         -          -
1/1/8 0100.c200.0003 10         11        v6in       v6out
1/1/9 0200.c200.0003 100        101       v6in       v6out
```

The following example displays information on ALCs for stack unit 1.

```
device# show authentication acls unit 1
-------------------------------------------------------------------------------
Port MAC Address     V4 Ingress V4 Egress V6 Ingress V6 Egress
-------------------------------------------------------------------------------
1/1/7 0180.c200.0003 -          -         -          -
1/1/8 0100.c200.0003 10         11        v6in       v6out
1/1/9 0200.c200.0003 100        101       v6in       v6out
```

The following example displays information on ACLs for port 1/1/9.

```
device# show authentication acls ethernet 1/1/9
-------------------------------------------------------------------------------
Port  MAC Address    V4 Ingress V4 Egress V6 Ingress V6 Egress
-------------------------------------------------------------------------------
1/1/9 0200.c200.0003 100        101       v6in       v6out
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# show authentication configuration

Displays the 802.1X and MAC authentication configuration for the device or the specified interface.

## Syntax

**show authentication configuration** [ **all** | **ethernet** { *unit* **/** *slot* **/** *port* } ]

## Parameters

**ethernet** { *unit* **/** *slot* **/** *port* }

Displays authentication configuration for the specified port.

**all**

Displays authentication configuration for all Flexible authentication ports.

## Modes

All modes.

## Usage Guidelines

The **show authentication configuration** command without parameters displays global configuration information for the ICX device.

# Examples

The following example displays configuration details for global authentication.

```
device# show auth configuration
Auth:
  Auth Order                    : mac-auth dot1x
  Default VLAN                  : 10
  Default Voice VLAN            : Not configured
  Auth Mode                     : Multiple Untagged Mode
  Restricted VLAN               : Not configured
  Critical VLAN                 : Not configured
  Auth Failure Action           : Block traffic
  Auth Timeout Action           : Treat as a successful authentication
  MAC Session Aging             : Enabled
  Re-authentication             : Enabled
  Reauth-period                 : 180 seconds
  Reauth-timeout                : 120 seconds
  Session Max SW-Age            : 120 seconds
  Session Aax HW-Age            : 70 seconds
  Max Sessions                  : 2
MAC-Auth:
  Status                        : Enabled
  802.1X Override               : Enabled
  Password Override             : Disabled
  Password Format               : xxxx.xxxx.xxxx
802.1X:
  Status                        : Enabled
  Protocol Version              : 1
  PAE Capability                : Authenticator Only
  MAC-Auth Override             : Disabled
  Guest VLAN                    : Not configured
  Quiet-period                  : 60 seconds
  TX-period                     : 30 seconds
  Supplicant-timeout            : 30 seconds
  Max Reauth Requests           : 2
  Max Frame Retries             : 2
```

The following example displays authentication configuration details for port 1/1/1.

```
device# show auth configuration ethernet 1/1/1
Port 1/1/1 Configuration:
  Auth Order                    : mac-auth dot1x
  Auth Mode                     : Multiple Untagged Mode
  Auth Failure Action           : Block traffic
  Auth Timeout Action           : Treat as a successful authentication
  DoS Protection                : Disabled (limit = 512)
  Source-guard Protection       : Disabled
  Aging                         : Enabled
  Max Sessions                  : 32
  Reauth-timeout                : 120 seconds
  802.1X Port-Control           : Auto
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# show authentication sessions

Displays details of 802.1X or MAC authentication sessions.

## Syntax

**show authentication sessions** { **all** | **ethernet** { *unit* **/** *slot* **/** *port* } | **unit** *unit_number* }

**show authentication sessions** { **brief** }

**show authentication sessions** { detail { **ethernet** { *unit* **/** *slot* **/** *port* } | **unit** *unit_number* } }

## Parameters

**all**

Displays information on all authentication sessions for the stack or standalone unit.

**ethernet** { *unit* **/** *slot* **/** *port* }

Displays authentication sessions for the specified interface or range of interfaces.

**unit** *unit_number*

Displays authentication sessions for the specified stack unit.

**brief**

Displays abbreviated information on authentication sessions.

**detail**

Displays detailed information on authentication sessions.

## Modes

Privileged EXEC or any configuration mode.

## Examples

The following example displays information on all authentication sessions for the stack.

```
device# show authentication sessions all
-------------------------------------------------------------------------------------------------------
-----
Port    MAC             IP(v4/v6)           User    VLAN Auth    Auth    ACL   Session  Age  PAE
        Addr            Addr                Name         Method  State         Time          State
-------------------------------------------------------------------------------------------------------
-----
2/1/25  00aa.aaaa.0000  198.1.1.2           MVDI_1 130  MAUTH   permit Yes   210     Ena   N/A
2/1/25  00aa.aaaa.0001  fe80::2aa:aaff:feaa DVDI_1 130  8021.X  permit Yes   210     Ena   AUTHENTICATED
                        3000::2
                        3000::2
1/1/15 00bb.bbbb.0001   N/A                 DVDI_2 230  8021.X  permit None  500    Ena    AUTHENTICATED
1/1/10 0010.9400.1101   N/A                 MVDI_2 330  MAUTH   permit None  410    Ena    N/A
```

The following example displays information on authentication sessions for a specific interface.

```
device(config)# show authentication sessions ethernet 2/1/25
-------------------------------------------------------------------------------------------------------------
-----
Port    MAC             IP(v4/v6)           User   VLAN Auth    Auth   ACL   Session  Age  PAE
        Addr            Addr                Name        Method  State        Time          State
-------------------------------------------------------------------------------------------------------------
-----
2/1/25  00aa.aaaa.0000  198.1.1.2           MVDI_1 130  MAUTH   permit Yes   210     Ena   N/A
2/1/25  00aa.aaaa.0001  fe80::2aa:aaff:feaa DVDI_1 130  8021.X  permit Yes   210     Ena   AUTHENTICATED
```

The following example displays session information for stack unit 1.

```
device(config)# show authentication sessions unit 1
-------------------------------------------------------------------------------------------------------------
-----
Port    MAC             IP(v4/v6)           User   VLAN Auth    Auth   ACL   Session  Age  PAE
        Addr            Addr                Name        Method  State        Time          State
-------------------------------------------------------------------------------------------------------------
-----
1/1/15  00bb.bbbb.0001  N/A                 DVDI_2 230  8021.X  permit None  500     Ena   AUTHENTICATED
1/1/10  0010.9400.1101  N/A                 MVDI_2 330  MAUTH   permit None  410     Ena   N/A
```

The following example displays a brief description of authentication sessions.

```
device# show authentication sessions brief
-----------------------------------------------------------------------------------------
Port   Number of         Number of          Number of      Untagged            Dynamic
       Attempted Users   Authorized Users   Denied Users   VLAN Type           Port ACL
       MAC DOT1X         MAC DOT1X          MAC DOT1X
-----------------------------------------------------------------------------------------
1/1/7  0   1             0   1              0   0          RADIUS-VLAN         No
1/1/8  1   0             1   0              0   0          Auth-Default-VLAN   No
1/1/9  0   0             0   0              0   0          Multiple No
1/1/10 0   0             0   0              0   0          Auth-Default-VLAN   No
```

The following example displays a detailed description of authentication sessions on port 17/1/1.

```
device# show authentication sessions detail ethernet 17/1/1
Auth Session Info (Port 17/1/1, MAC a036.9f6e.1fd2) :
State                 : Permitted
Auth Method           : 802.1X              Auth Mode              : Single Untagged
VLAN Type             : Radius-VLAN         VLAN                   : 200
Voice VLAN            : 0                   PVID                   : 0
Tagged VLANs          :
User Name             : joe.user@arris.com
Session Time          : 1381                Reauth Time            : 2220
Idle Timeout          : 120                 Session Timeout        : 0
Acct session ID       : 2                   PCE Index              : 65535
PAE State             : AUTHENTICATED       Age                    : Disabled
Qos Priority          : 0                   Failure Reason         :
Auth Filter Applied   : No                  Tagged                 : No
VLAN Add Req State    : Complete            VLAN Del Req State     : Init
Filter Add Req State  : Complete            Filter Del Req State   : Init
Stale                 : No                  Delete Pending         : No
802.1X Enabled        : No                  Session Control        : Self
V4 ACL Applied        : No                  V6 ACL Applied         : No
V4 IN ACL (Session)   : acl1                V4 OUT ACL (Session)   : -
V6 IN ACL (Session)   : -                   V6 OUT ACL (Session)   : -
Client Voice Phone    : No                  Client Wireless AP     : No
802.1X Capable        : Yes
IP Addresses          : 10.176.167.145
V4-IN ACL (Dynamic)   : 3928                V4-OUT ACL (Dynamic)   : 0
V6-IN ACL (Dynamic)   : 0                   V6-OUT ACL (Dynamic)   : 0
V4-IN ACL RefCnt      : 1                   V4-OUT ACL RefCnt      : 0
V6-IN ACL RefCnt      : 0                   V6-OUT ACL RefCnt      : 0
V4 ACL Trap Rule      : Yes                 V6 ACL Trap Rule       : No
Addr Change Count     : 0                   MBV Usage Count        : 1
Radius VLAN RefCnt    : 1
Auth Order            : dot1x, mac-auth     Auth Fail Action       : Restricted VLAN (3)
Auth Timeout Action   : Failure             Aging : Enabled
SG Protection         : Disabled            DOS Protection         : Disabled (limit = 512)
Reauthentication      : Enabled             Reauth Period          : 3600
Reauth Timeout        : 300                 Max Ssessions          : 2
Port Control          : Auto                Quiet Period           : 60
Supplicant Time       : 30                  Tx Period              : 3
Max Reauth Requests   : 2 Max               Frame Retries          : 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# show authentication statistics

Displays authentication statistics for a specified interface.

## Syntax

**show authentication statistics** { **all** | **ethernet** { *unit* **/** *slot* **/** *port* } | **unit** *unit_number* }

## Parameters

**all**

Displays information on all authentication statistics for the stack or standalone unit.

**ethernet** *unit* **/** *slot* **/** *port*

Displays authentication statistics for the specified port.

**unit** *unit_number*

Displays authentication statistics for the specified stack unit.

## Modes

All modes.

## Command Output

The **show authentication statistics** command displays the following information:

| Output field | Description |
|---|---|
| RX EAPOL Start | The number of EAPOL-Start frames received on the port |
| RX EAPOL Logoff | The number of EAPOL-Logoff frames received on the port |
| RX EAPOL Invalid | The number of invalid EAPOL frames received on the port |
| RX EAPOL Total | The total number of EAPOL frames received on the port |
| RX EAP Resp/Id | The number of EAP-Response/Identity frames received on the port |
| RX EAP Resp other than Resp/Id | The total number of EAPOL-Response frames received on the port that were not EAP-Response/Identity frames |
| RX EAP Length Error | The number of EAPOL frames received on the port that have an invalid packet body length |
| Last EAPOL Version | The version number of the last EAPOL frame received on the port |
| Last EAPOL Source | The source MAC address in the last EAPOL frame received on the port |
| TX EAPOL Total | The total number of EAPOL frames transmitted on the port |
| TX EAP Req/Id | The number of EAP-Request/Identity frames transmitted on the port |
| TX EAP Req other than Req/Id | The number of EAP-Request frames transmitted on the port that were not EAP-Request/Identity frames |
| Accepted Sessions | The number of MAC authentication sessions accepted |
| Rejected Sessions | The number of MAC authentication sessions rejected |
| In Progress Sessions | The number of MAC authentication sessions in progress |
| Attempted Sessions | The number of MAC authentication sessions attempted |

| Output field | Description |
|---|---|
| Number of Errors | The number of errors encountered while processing sessions |

# Examples

The following example displays authentication statistics for port 1/1/1.

```
device# show authentication statistics ethernet 1/1/1
Port 1/1/1 Statistics:
  802.1X:
    RX EAPOL Start:                 0
    RX EAPOL Logoff:                0
    RX EAPOL Invalid:               0
    RX EAPOL Total:                 0
    RX EAP Resp/Id:                 0
    RX EAP Resp other than Resp/Id: 0
    RX EAP Length Error:            0
    Last EAPOL Version:             0
    Last EAPOL Source:              0000.0050.0B83
    TX EAPOL Total:                 217
    TX EAP Req/Id:                  163
    TX EAP Req other than Req/Id:   0
  MAC-Auth:
    Accepted Sessions:              0
    Rejected Sessions:              0
    Inprogress Sessions:            0
    Attempted Sessions:             0
    Number of Errors:               0
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# show boot-monitor

This CLI displays the current and the recommended boot-monitor version, allowing you to understand the mismatch.

## Syntax

**show boot-monitor**

## Modes

Privileged EXEC mode

## Usage Guidelines

Whenever current boot-version is not same as the recommended boot-monitor version, show version command displays alert message for you to indicate the mismatch in boot-monitor version and prompts you to run the **show boot-monitor** command.

## Examples

The following is an example of the output displayed from the **show boot-monitor** command.

```
device#show boot-monitor
UNIT1:
Current Version: 10.1.11b006
Recommended Version: 10.1.11b014(Mismatch)
UNIT2:
Current Version: 10.1.11b014
Recommended Version: 10.1.11b014
UNIT3:
Current Version: 10.1.11b014
Recommended Version: 10.1.11b014
UNIT4:
Current Version: 10.1.11b006
Recommended Version: 10.1.11b014 (Mismatch)
```

# show batch schedule

Displays the schedule and status of batch execution.

## Syntax

**show batch schedule**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Examples

The following example displays the status and schedule of batch buffer execution.

```
device# show batch schedule
Printing the details of Timer
Batch buffer 1 timer is off
Batch buffer 2 timer is off
Batch buffer 3 timer is off
Batch buffer 4 timer is off
Printing Details of Start Timer
Batch buffer 1  start timer will be executed 0 days 0 hours 4 minutes 20 seconds from now
Batch buffer 2 start timer is off
Batch buffer 3 start timer is off
Batch buffer 4 start timer is off
Printing Details of Stop Timer
Batch buffer 1 stop timer will be executed 9 days 20 hours 44 minutes 19 seconds from now
Batch buffer 2 stop timer is off
Batch buffer 3 stop timer is off
Batch buffer 4 stop timer is off
```

# show boot-preference

Displays the boot sequence in the startup configuration and running configuration files.

## Syntax

**show boot-preference**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

All configuration modes

## Usage Guidelines

The displayed boot sequence is also identified as user-configured or the default.

## Examples

The following example shows the default boot sequence preference.

```
device# show boot-preference

Boot system preference (Configured):
            Use Default
Boot system preference(Default):
            Boot system flash primary
            Boot system flash secondary
```

The following example shows a user-configured boot sequence preference.

```
device# show boot-preference

Boot system preference(Configured):
            Boot system flash primary
Boot system preference(Default):
            Boot system flash primary
            Boot system flash secondary
```

# show breakout

Displays information on 10 Gbps sub-ports broken out from 40 Gbps ports on certain FastIron devices.

## Syntax

**show breakout**

## Modes

Privileged EXEC mode.

## Usage Guidelines

The **show breakout** command is available only on ICX 7750 devices.

## Command Output

The **show breakout** command displays the following information:

| Output field | Description |
|---|---|
| Port | Specifies the port for which breakout information is displayed to the right. |
| Module Exist | Indicates whether the module on which the specified port resides is present in the unit. |
| Module Conf | Indicates whether the module on which the specified port resides is configured. |
| Breakout-config | Indicates whether breakout is configured on the specified port. |
| Breakout-oper | Indicates whether sub-ports on the specified breakout port are operational. |

## Examples

The following example shows that port 1/2/1 has been configured for breakout into four 10 Gbps sub-ports and is operational (has active sub-ports). Ports 1/2/2 and 1/2/4 are configured for breakout, pending reload.

```
Device# show breakout
Unit-Id: 1
Port         Module Exist  Module Conf    Breakout-config   Breakout-oper
1/2/1        yes           no             yes               yes
1/2/2        yes           no             yes               no
1/2/3        yes           no             no                no
1/2/4        yes           no             yes               no
1/2/5        yes           no             no                no
1/2/6        yes           no             no                no
1/3/1        yes           no             no                no
1/3/2        yes           no             no                no
1/3/3        yes           no             no                no
1/3/4        yes           no             no                no
1/3/5        yes           no             no                no
1/3/6        yes           no             no                no
```

# History

| Release version | Command history |
|---|---|
| FastIron Release 08.0.30 | This command was introduced. |

# show cable-diagnostics tdr

Displays the results of Virtual Cable Test (VCT) TDR cable diagnostic testing.

## Syntax

**show cable-diagnostics tdr** *stackid**/**slot**/ port*

## Parameters

***stackid/slot/port*** Identifies the specific interface (port), by device, slot, and port number in the format shown.

## Modes

User EXEC mode

Privileged EXEC mode

## Usage Guidelines

Most Ruckus ICX devices support VCT technology. VCT technology enables the diagnosis of a conductor (wire or cable) by sending a pulsed signal into the conductor, then examining the reflection of that pulse. This method of cable analysis is referred to as Time Domain Reflectometry (TDR). By examining the reflection, the device can detect and report cable statistics such as local and remote link pair, cable length, and link status.

## Examples

The following example displays TDR test results for port 1, slot 2 on device 3 in the stack. The results indicate that the port is down or the cable is not connected.

```
device>show cable-diagnostics tdr 3/2/1

Port      Speed Local pair Pair Length Remote pair Pair status
--------- ----- ---------- ----------- ----------- -----------
01        UNKWN Pair A     >=3 M                   Open
                Pair B     >=3 M                   Open
                Pair C     >=3 M                   Open
                Pair D     >=3 M                   Open
```

The following example displays I the TDR test results for the same port show details for an active port.

```
device>show cable-diagnostics tdr 3/2/1

Port      Speed Local pair Pair Length Remote pair Pair status
--------- ----- ---------- ----------- ----------- -----------
01        1000M Pair A     50M         Pair B      Terminated
                Pair B     50M         Pair A      Terminated
                Pair C     50M         Pair D      Terminated
                Pair D     50M         Pair C      Terminated
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show captive-portal

Displays the details of the Captive Portal profile configured on the device.

## Syntax

**show captive-portal** *profile-name*

## Parameters

*profile-name*

Specifies a specific Captive Portal profile configured on the device.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Web Authentication configuration mode

## Command Output

The **show captive-portal** command displays the following information:

| Output field | Description |
|---|---|
| cp-name | The local account username. |
| virtual-ip | Captive portal server name or ip address. |
| virtual-port | The port number to facilitate HTTP services for the client. The port can be secure HTTPS port 443 or unsecure HTTP port 80. |
| login-page | The login-page hosted on the external web server. |

## Examples

The following example displays the details for the cp-ruckus Captive Portal profile. The external web server used in this case is a Ruckus Cloudpath.

```
device(config)# show captive-portal cp-ruckus
Configured Captive Portal Profile Details :
  cp-name               :cp-ruckus
  virtual-ip            :10.21.240.50
  virtual-port          :80
  login-page            :/enroll/ruckus/guestlogin.php
```

The following example displays the details for the cp-ruckus Captive Portal profile. The external web server used in this case is an Aruba Clearpass.

```
device(config)# show captive-portal cp-ruckus
Configured Captive Portal Profile Details :
  cp-name               :cp-ruckus
  virtual-ip            :10.21.240.42
  virtual-port          :80
  login-page            :/guest/ruckus/guestlogin.php
```

The following example displays the details for the cp-ruckus Captive Portal profile. The external web server used in this case is a Cisco ISE.

```
device(config)# show captive-portal cp-ruckus
Configured Captive Portal Profile Details :
  cp-name               :cp-ruckus
  virtual-ip            :10.21.240.48
  virtual-port          :80
  login-page            :ruckusguestlogin
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j. |

# show chassis

Displays chassis information.

## Syntax

**show chassis**

## Command Output

The **show chassis** command output displays the following information.

| Output field | Description |
|---|---|
| Power supply # | The presence, status, output type, model number, serial number, and firmware version number of the power supply units, if present. |
| Power supply # Fan Air Flow Direction | The air flow direction of the power supply unit. |
| Fan # | The presence, status, speed mode, and air flow direction of the fan. The fan controlled temperature and temperature thresholds. |
| MAC # Temperature Readings | The current temperature reading of the MAC device. |
| CPU Temperature Readings | The current temperature reading of the CPU. |
| sensor # Temperature Readings | The current temperature reading of the sensor. |
| Boot Prom MAC | The MAC address of the boot prom. |
| Management MAC | The management MAC address, for the active controller only. |

## Modes

Privileged EXEC mode

# Examples

The following is sample output from the **show chassis** command executed on an ICX 7450 device.

```
device# show chassis
The stack unit 1 chassis info:

Power supply 1 not present
Power supply 2 (AC - Regular) present, status ok
        Model Number:   23-0000144-02
        Serial Number:  07W
        Firmware Ver:    A
Power supply 2 Fan Air Flow Direction:  Front to Back

Fan 1 not present
Fan 2 ok, speed (auto): [[1]]<->2

Fan controlled temperature: 40.5 deg-C

Fan speed switching temperature thresholds:
                Speed 1: NM<----->66       deg-C
                Speed 2:       56<-----> 85 deg-C (shutdown)

Fan 2 Air Flow Direction:  Front to Back
Slot 1 Current Temperature: 31.0 deg-C (Sensor 1), 41.0 deg-C (Sensor 2), 38.5 d
eg-C (Sensor 3), 29.5 deg-C (Sensor 4)
Slot 2 Current Temperature: 31.0 deg-C (Sensor 1)
Slot 3 Current Temperature: 31.0 deg-C (Sensor 1)
Slot 4 Current Temperature: 31.5 deg-C (Sensor 1)
        Warning level.......: 70.0 deg-C
        Shutdown level......: 85.0 deg-C
Boot Prom MAC : cc4e.248b.b050
Management MAC: cc4e.248b.b050
```

The following is sample output from the **show chassis** command executed on ICX 7150-24P, ICX 7150-48P, or ICX 7150-48PF devices.

```
device# show chassis

The stack unit 1 chassis info:

Power supply 1 (AC - PoE) present, status ok

Fan 1 ok, speed (auto): [[1]]<->2
Fan 2 ok, speed (auto): [[1]]<->2

Fan controlled temperature:
        Rule 1/2 (MGMT THERMAL PLANE): 43.2 deg-C
        Rule 2/2 (AIR OUTLET NEAR PSU): 28.0 deg-C

Fan speed switching temperature thresholds:
        Rule 1/2 (MGMT THERMAL PLANE):
                Speed 1: NM<-----> 70       deg-C
                Speed 2:        60<----->105 deg-C (shutdown)
        Rule 2/2 (AIR OUTLET NEAR PSU):
                Speed 1: NM<-----> 58       deg-C
                Speed 2:        49<----->105 deg-C (shutdown)

Fan 1 Air Flow Direction:Front to Back
Fan 2 Air Flow Direction:Front to Back
Slot 1 Current Temperature: 43.7 deg-C (Sensor 1), 43.2 deg-C (Sensor 2), 28.0 deg-C (Sensor 3), 36.3
deg-C (Sensor 4), 34.2 deg-C (Sensor 5)
Slot 2 Current Temperature: NA
Slot 3 Current Temperature: NA
        Warning level.......: 100.0 deg-C
        Shutdown level......: 105.0 deg-C
Boot Prom MAC : 609c.9ffc.3b7c
```

The following is sample output from the **show chassis** command executed on an ICX 7150-48ZP device.

```
device# show chassis

The stack unit 1 chassis info:

Power supply 1 present, status failed, reason NO AC POWER INPUT
Power supply 2 (AC - PoE) present, status ok
        Model Number:   YM-1921AB06R
        Serial Number:  SA000V171708000163
        Firmware Ver:   P2H802A00
Power supply 2 Fan Air Flow Direction:  Front to Back

Fan 1 ok, speed (auto): [[1]]<->2
Fan 2 ok, speed (auto): [[1]]<->2

Fan controlled temperature:
        Rule 1/3 (MGMT THERMAL PLANE): 35.9 deg-C
        Rule 2/3 (PoE THERMAL SENSOR PLANE): 35.0 deg-C
        Rule 3/3 (MISC THERMAL PLANE): 51.0 deg-C

Fan speed switching temperature thresholds:
        Rule 1/3 (MGMT THERMAL PLANE):
                Speed 1: NM<-----> 95       deg-C
                Speed 2:        85<----->105 deg-C (shutdown)
        Rule 2/3 (PoE THERMAL SENSOR PLANE):
                Speed 1: NM<----->130       deg-C
                Speed 2:       120<----->130 deg-C
        Rule 3/3 (MISC THERMAL PLANE):
                Speed 1: NM<----->100       deg-C
                Speed 2:        85<----->108 deg-C

Fan 1 Air Flow Direction:  Front to Back
Fan 2 Air Flow Direction:  Front to Back
Slot 1 Current Temperature: 33.0 deg-C (Sensor 1), 36.4 deg-C (Sensor 2), 35.0 deg-C (Sensor 3), 50.0
deg-C (Sensor 4)
Slot 2 Current Temperature: NA
        Warning level.......: 102.0 deg-C
        Shutdown level......: 105.0 deg-C
Boot Prom MAC : 609c.9fe2.12ce
```

# History

| Release | Command History |
|---|---|
| 08.0.00a | This command was enhanced to display model number, serial number, and firmware version number for power supply units. |
| 08.0.60 | The command output was enhanced to display the status of the fan as fanless mode if the mode is enabled. |
| 08.0.80 | The command output was enhanced to display the temperature of the air outlet near the PSU for ICX 7150-24P, ICX 7150-48P, or ICX 7150-48PF devices. |

# show cli-command-history

Displays the history list of CLI commands executed on the device from any console, Telnet, or SSH session.

## Syntax

**show cli-command-history** [ **wide** ]

## Parameters

**wide**

Displays the complete form of the command names that are truncated in the output.

## Modes

User EXEC mode

Global configuration

## Command Output

The **show cli-command-history** command displays the following information:

| Output field | Description |
|---|---|
| Session | The session type from which the command was executed. |
| User-name | The local account username. |
| Ip-address | The IP address of the device. |
| Executed-time | The time at which the command was executed. |
| Command | The command that was executed. |

## Examples

The following example shows the history list of commands executed on the device.

```
device# show cli-command-history

Slno Session  User-name            Ip-address   Executed-time   Command
   1 console  Un-authenticated user             Jun  2 10:15:54 no crypto-ssl certificate zero*
   2 console  Un-authenticated user             Jun  2 10:15:42 show files
   3 console  Un-authenticated user             Jun  2 10:15:39 show web
   4 console  Un-authenticated user             Jun  2 10:15:36 no web-management http
   5 console  Un-authenticated user             Jun  2 10:15:20 show web
   6 console  Un-authenticated user             Jun  2 10:14:53 write memory
  36 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show ip
  37 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show dir
  38 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show users
  39 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show files
  40 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show version
  41 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show ip ssh
  42 telnet_5 Ruckus               10.70.43.98 Jun  2 09:46:06 show ip address
```

The following example shows the complete form of the commands executed on the device.

```
device(config)# show cli-command-history wide

Slno Session User-name            Ip-address   Executed-time   Command
   1 console Un-authenticated user             Jun  2 10:15:54 no crypto-ssl certificate zeroize
   2 console Un-authenticated user             Jun  2 10:15:42 show files
   3 console Un-authenticated user             Jun  2 10:15:39 show web
   4 console Un-authenticated user             Jun  2 10:15:36 no web-management http
   5 console Un-authenticated user             Jun  2 10:15:20 show web
   6 console Un-authenticated user             Jun  2 10:14:53 write memory
  36 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show ip
  37 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show dir
  38 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show users
  39 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show files
  40 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show version
  41 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show ip ssh
  42 telnet_5 Ruckus              10.70.43.98 Jun  2 09:46:06 show ip address
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show clock

Displays the local time, date, and time zone.

## Syntax

**show clock** [ **detail** ]

## Parameters

**detail**

Displays detailed information for the system clock inlcuding the summer-time settings.

## Modes

Privileged EXEC mode

## Usage Guidelines

To set the local system clock you can configure the date and time using the **clock set** command. Use the **clock timezone** command to set the timezone. If the daylight savings time is different from the default timezone setting, use the optional **clock summer-time** command to set the time and date for the start and end of the daylight savings period.

## Examples

In the following example, the local system clock time, timezone, date, and time source are displayed.

```
device# show clock

03:35:53.658 Mountain Wed Aug 03 2016
Time source is Set Clock
```

In the following example, the local system clock time, timezone, date, time source, and summer time start and end dates and times are displayed.

```
device# show clock detail

03:35:53.658 Mountain Wed Aug 03 2016
Time source is Set Clock
Summer time starts 02:00:00 Mountain Sun Feb 28 2016 offset 30 mins
Summer time ends 02:00:00 Mountain Sun Oct 30 2016 offset 30 mins
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was modified to display additional subsets of time zones specific to Australia and Europe and to display the optional offset value. |

# show configuration

Displays the configuration data in the startup configuration file.

## Syntax

**show configuration**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

All configuration modes

## Examples

The following example is sample output from the **show configuration** command.

```
device# show configuration
!
Startup-config data location is flash memory
!
Startup configuration:
!
ver 08.0.20
!
stack unit 1
  module 1 icx7450-24-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 3 icx7400-qsfp-1port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
!
!
!
!
!
!
!
boot sys fl sec
ip address 10.25.224.197 255.255.255.0 dynamic
ip dns domain-list englab.ruckus.com
ip dns server-address 10.31.2.10
ip default-gateway 10.25.224.1
!
!
!
!
!
!
!
!
!
!
end
```

The following example is sample output from the **show configuration** command for a switch, including dynamically obtained DHCP options.

```
device> show configuration

Startup-config data location is flash memory
!
Startup configuration:
!
ver 08.0.61b1T211
!
stack unit 1
  module 1 icx7250-24-port-management-module
  module 2 icx7250-sfp-plus-8port-80g-module
!
!
vlan 1 name DEFAULT-VLAN by port
!
!
!
!
ip address 10.10.10.2 255.255.255.0 dynamic
ip dns domain-list ManualDomain.com
ip dns domain-list testStaticDomain.com
ip dns server-address 20.20.20.8 20.20.20.9 20.20.20.5
ip default-gateway 10.10.10.1 dynamic
!
!
!
interface ethernet 1/1/21
 disable
!
interface ethernet 1/2/2
 speed-duplex 1000-full
!
interface ethernet 1/2/4
 speed-duplex 1000-full
!
interface ethernet 1/2/5
 speed-duplex 1000-full
!
interface ethernet 1/2/6
 speed-duplex 1000-full
!
interface ethernet 1/2/7
 speed-duplex 1000-full
!
interface ethernet 1/2/8
 speed-duplex 1000-full
!
lldp run
!

end
```

The following example is sample output from the **show configuration** command for a router, including dynamically obtained DHCP options.

```
device> show configuration

!
Startup-config data location is flash memory
!
Startup configuration:
!
ver 08.0.61b1T213
!
stack unit 1
  module 1 icx7250-24-port-management-module
  module 2 icx7250-sfp-plus-8port-80g-module
!
!
vlan 1 name DEFAULT-VLAN by port
!
!
!
ip dns domain-list ManualDomain.com
ip dns domain-list testStaticDomain.com
ip dns server-address 20.20.20.8 20.20.20.9 20.20.20.5
ip route 0.0.0.0/0 10.10.10.1 distance 254 dynamic
!
!
!
interface ethernet 1/1/7
 ip address 10.10.10.2 255.255.255.0 dynamic
!
interface ethernet 1/1/21
 disable
!
interface ethernet 1/2/2
 speed-duplex 1000-full
!
interface ethernet 1/2/4
 speed-duplex 1000-full
!
interface ethernet 1/2/5
 speed-duplex 1000-full
!
interface ethernet 1/2/6
 speed-duplex 1000-full
!
interface ethernet 1/2/7
 speed-duplex 1000-full
!
interface ethernet 1/2/8
 speed-duplex 1000-full
!
!
lldp run
!
!
end
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include information about dynamically obtained DHCP options. |

# show configuration (SPX)

Shows the startup configuration in regular switch or router mode, or the PE startup configuration in Provisional-PE or PE mode.

## Syntax

**show configuration**

## Modes

Device mode

PE mode

Provisional-PE mode

## Usage Guidelines

In regular switch or router mode, the **show configuration** command shows the saved startup configuration.

In PE or Provisional-PE mode, the **show configuration** command shows the configuration in the PE startup file for this unit. To view the configuration that the unit would have in regular mode (as a switch or router), use the **show startup-config** command.

## Command Output

The **show configuration** command displays the following information in Provisional-PE and PE mode:

| Output field | Description |
|---|---|
| ver | Software version loaded to this unit. |
| spx-enable | The unit has been enabled to act as an SPX PE unit. |
| spx unit *x* | SPX unit number. Locally, the SPX number will always be 1 for the unit. |
| module *x* | Module number and type installed for this SPX unit. |
| spx-lag | SPX LAG configuration for this unit (if any). |
| spx-port | SPX port configuration for this unit (if any). |

# Examples

The following example shows the saved startup configuration for a provisional PE that has been enabled with the **spx pe-enable** command.

```
[Provisional-PE]device# show configuration
Configuration in PE startup file:
!
ver 08.0.40b1T213
!
spx pe-enable
spx unit 1
  module 1 icx7450-48f-sf-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
  spx-lag 1/2/1 to 1/2/2
  spx-port 1/2/4
```

The following example shows the startup configuration for the active PE unit. Jumbo mode has been enabled on the CB.

```
[PE]local-id@device# show configuration
Configuration in PE startup file:
!
ver 08.0.40b728T213
!
spx pe-enable
spx unit 1
  module 1 icx7450-48p-poe-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 3 icx7400-qsfp-1port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
  spx-lag 1/1/1 to 1/1/3
  spx-port 1/2/4
!
jumbo
!
[PE]local-id@device#
```

The following example shows command output when the PE configuration has not been saved with the **write memory** command.

```
[Provisional-PE]device(config-spx-unit-1)# show configuration
PE startup file does not exist.
[Provisional-PE]device(config-spx-unit-1)# end
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show cpu histogram

Displays the CPU usage histogram for the device, and optionally, clears the hold time and wait time.

## Syntax

**show cpu histogram** [ **clear** | **holdtime** | **waittime** ]

## Parameters

**clear**

Displays the CPU usage histogram and clears the hold time and wait time.

**holdtime**

Displays the CPU hold time usage histogram.

**waittime**

Displays the CPU wait time usage histogram.

## Modes

Global configuration mode

User EXEC

## Usage Guidelines

## Command Output

The **show cpu histogram** command displays the following information:

| Output field | Description |
|---|---|
| No. of buckets | The CPU usage histogram is presented in the form of buckets. Usage is divided into different intervals called buckets. |
| Bucket granularity | The time interval at which the CPU usage information is collected for each bucket. |
| Last clear | The datestamp when the task was cleared last. |

# Examples

The following command displays the CPU hold time usage histogram.

```
device# show cpu histogram holdtime

CPU Histogram Info
------------------
No. of Buckets     : 11
Bucket Granularity : 50 msec
No. of Tasks       : 14
Last clear         : Jan  1 18:11:39.414

-------------------------------------------------------------------------------------------------------
 Task                |   Bkt |  Bkt     | Total     | Last         | Max          |     Max Hold at    |
 Name                |   Num |  Time(ms)| Count     | HoldTime(ms)| HoldTime(ms)|
                     |
-------------------------------------------------------------------------------------------------------
appl                        1     000-050  758226345         9.521        46.543
Jan  1 18:50:16.857
appl                        2     050-100          4        50.967        52.324
Jan  1 18:46:00.638
rtm                         1     000-050      44197         0.008         0.283
Jan  1 18:33:37.651
rtm6                        1     000-050      44197         0.005         0.415
Jan  1 18:18:31.476
ospf                        1     000-050      44197         0.004         1.177
Jan  1 19:02:29.746
openflow_opm                1     000-050       9118         0.007         0.239
Jan  1 18:15:01.952
mcast                       1     000-050      90565         0.004         0.143
Jan  1 18:29:04.325
msdp                        1     000-050       4425         0.007         0.201
Jan  1 19:15:34.419
ospf6                       1     000-050      44197         0.007         0.257
Jan  1 18:44:58.033
mcast6                      1     000-050      90565         0.004         0.181
Jan  1 18:36:38.346
rmon                        1     000-050       4425         0.028         5.787
Jan  1 19:24:47.464
web                         1     000-050      88335         0.010         0.368
Jan  1 18:29:48.222
acl                         1     000-050       2360         0.015         0.177
Jan  1 18:22:40.049
ntp                         1     000-050       4425         0.007         0.011
Jan  1 18:11:40.713
console                     1     000-050      88337         0.008        35.227
Jan  1 18:11:39.498
-------------------------------------------------------------------------------
--------------------
```

The following example displays the CPU wait time usage histogram.

```
device# show cpu histogram waittime
CPU Histogram Info
------------------
No. of Buckets     : 11
Bucket Granularity : 50 msec
No. of Tasks       : 14
Last clear         : Jan  1 18:11:39.414

-------------------------------------------------------------------------------------------------------
 Task                |   Bkt |  Bkt     | Total     | Last         | Max          |     Max Wait at   |
 Name                |   Num |  Time(ms)| Count     | WaitTime(ms)| WaitTime(ms)|
                     |
-------------------------------------------------------------------------------------------------------
rtm                        1     000-050      44876         0.008         0.283
Jan  1 18:50:16.857
rtm6                       1     000-050      44876         0.005         0.415
Jan  1 18:50:16.857
ospf                       1     000-050      44876         0.065         1.177
Jan  1 18:50:16.857
openflow_opm               1     000-050       9258         0.006         0.239
Jan  1 19:07:56.599
mcast                      1     000-050      91957         0.005         0.143
Jan  1 18:50:16.857
msdp                       1     000-050       4493         0.008         0.201
Jan  1 18:28:40.956
ospf6                      1     000-050      44876         0.007         0.257
Jan  1 18:50:16.857
mcast6                     1     000-050      91957         0.004         0.181
Jan  1 18:50:16.857
rmon                       1     000-050       4493         0.030         5.787
Jan  1 18:28:40.956
web                        1     000-050      89691         0.009         0.368
Jan  1 18:50:16.857
acl                        1     000-050       2397         0.018         0.177
Jan  1 18:33:17.172
ntp                        1     000-050       4493         0.007         0.011
Jan  1 18:28:40.956
console                    1     000-050      89693         0.010        35.227
Jan  1 18:50:16.857
-------------------------------------------------------------------------------
```

The following example clears the CPU usage histogram information.

```
device# show cpu histogram clear

CPU Histogram Info
------------------
No. of Buckets     : 11
Bucket Granularity : 50 msec
No. of Tasks       : 14
Last clear         : Jan  1 18:11:39.414

-------------------------------------------------------------------------------------------------------
 Task                |   Bkt |  Bkt     | Total     | Last         | Max          |     Max Hold at    |
 Name                |   Num |  Time(ms)| Count     | HoldTime(ms)| HoldTime(ms)|
                     |
-------------------------------------------------------------------------------------------------------
appl                       1     000-050   793262215        0.003        46.543
Jan  1 18:50:16.857
appl                       2     050-100           4       50.967        52.324
Jan  1 18:46:00.638
rtm                        1     000-050       46242         0.009         0.283
Jan  1 18:33:37.651
rtm6                       1     000-050       46242         0.005         0.415
Jan  1 18:18:31.476
ospf                       1     000-050       46242         0.006         1.177
Jan  1 19:02:29.746
openflow_opm               1     000-050        9540         0.007         0.239
Jan  1 18:15:01.952
mcast                      1     000-050       94771         0.003         0.143
Jan  1 18:29:04.325
msdp                       1     000-050        4629         0.008         0.201
Jan  1 19:15:34.419
ospf6                      1     000-050       46242         0.006         0.257
Jan  1 18:44:58.033
mcast6                     1     000-050       94771         0.003         0.181
Jan  1 18:36:38.346
rmon                       1     000-050        4629         0.137         5.787
Jan  1 19:24:47.464
web                        1     000-050       92421         0.007         0.368
Jan  1 18:29:48.222
acl                        1     000-050        2470         0.006         0.177
Jan  1 18:22:40.049
ntp                        1     000-050        4629         0.006         0.011
Jan  1 18:11:40.713
console                    1     000-050       92423         0.008        35.227
Jan  1 18:11:39.498
-------------------------------------------------------------------------------------------------------

CPU Histogram Info
------------------
No. of Buckets     : 11
Bucket Granularity : 50 msec
No. of Tasks       : 14
Last clear         : Jan  1 18:11:39.414

-------------------------------------------------------------------------------------------------------
 Task                |   Bkt |  Bkt     | Total     | Last         | Max          |     Max Wait at    |
 Name                |   Num |  Time(ms)| Count     | WaitTime(ms)| WaitTime(ms)|
                     |
-------------------------------------------------------------------------------------------------------
rtm                        1     000-050       46242         0.009         0.283
Jan  1 18:50:16.857
rtm6                       1     000-050       46242         0.005         0.415
Jan  1 18:50:16.857
ospf                       1     000-050       46242         0.006         1.177
Jan  1 18:50:16.857
openflow_opm               1     000-050        9540         0.007         0.239
Jan  1 19:07:56.599
mcast                      1     000-050       94771         0.003         0.143
Jan  1 18:50:16.857
msdp                       1     000-050        4629         0.008         0.201
```

```
Jan  1 18:28:40.956
ospf6                    1     000-050      46242         0.006         0.257
Jan  1 18:50:16.857
mcast6                   1     000-050      94771         0.003         0.181
Jan  1 18:50:16.857
rmon                     1     000-050       4629         0.137         5.787
Jan  1 18:28:40.956
web                      1     000-050      92421         0.007         0.368
Jan  1 18:50:16.857
acl                      1     000-050       2470         0.006         0.177
Jan  1 19:28:22.095
ntp                      1     000-050       4629         0.006         0.011
Jan  1 18:28:40.956
console                  1     000-050      92423         0.008        35.227
Jan  1 18:50:16.857
-------------------------------------------------------------------------------------------------
CPU Histogram data cleared
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show cpu-utilization

Displays the CPU histogram for the device, and optionally, the CPU utilization for each task running on the device.

## Syntax

**show cpu** [ **tasks** ]

## Parameters

**tasks**

Specifies the display of CPU utilization information for each task running on the device.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

## Examples

The following example displays the CPU histogram for the device.

```
device# show cpu-utilization
cpu0:
1 percent busy, from 8213 sec ago
1   sec avg:  1 percent busy
5   sec avg:  1 percent busy
60  sec avg:  1 percent busy
300 sec avg:  1 percent busy
cpu1:
0 percent busy, from 7402 sec ago
1   sec avg: 0 percent busy
5   sec avg: 0 percent busy
60  sec avg: 0 percent busy
300 sec avg: 0 percent busy
```

The following example displays the CPU utilization for each task on the device.

```
device# show cpu tasks
... Usage average for all tasks in the last 1 second ...
============================================================
Name                                    %

SigHdlrTsk                              0
OsTsk                                   0
TimerTsk                                0
FlashTsk                                0
MainTsk                                 0
MportPollTsk                            0
IntrTsk                                 0
keygen                                  0
itc                                     0
bcmDPC                                  0
bcmINTR                                 3
socdmadesc.0                            0
bcmCNTR.0                               3
bcmTX                                   0
bcmXGS3AsyncTX                          0
bcmRX                                   0
bcmL2MOD.0                              0
scp                                     0
appl                                    86
snms                                    0
rtm                                     0
rtm6                                    0
rip                                     0
bgp                                     0
bgp_io                                  0
ospf                                    0
ospf_r_calc                             0
mcast_fwd                               0
mcast                                   0
msdp                                    0
ripng                                   0
ospf6                                   0
ospf6_rt                                0
mcast6                                  0
ipsec                                   0
dhcp6                                   0
snmp                                    0
rmon                                    0
web                                     0
acl                                     0
ntp                                     0
console                                 0
ospf_msg_task                           0
auxTsk                                  0
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was introduced. |
| 8.0.50 | The command output was modified to show information for cpu0 and cpu1. |

# show errdisable

Displays information about errdisabled ports.

## Syntax

**show errdisable** { **recovery** | **summary** }

## Parameters

**recovery**

Displays all the default error disable recovery states for all possible conditions.

**summary**

Displays the port number along with the reason why the port is in an errdisable state and the method used to recover the port.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

# Examples

The following example shows the errdisable recovery information.

```
device# show errdisable recovery
ErrDisable Reason                                                    Timer Status
---------------------------------------------------------------------------------
all reason                                                           Enabled
bpduguard                                                            Enabled
loopDetection                                                        Enabled
invalid license                                                      Disabled
packet-inerror                                                       Enabled
Reload the switch or stack to enable this port in 10G speed          Disabled
stack-port-resiliency                                                Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
                                                                     Disabled
loam-critical-event                                                  Disabled
                                                                     Disabled
                                                                     Disabled

Timeout Value: 300 seconds
PoD Timeout Value: 30 seconds

Interface that will be enabled at the next timeout:

Interface         Errdisable reason   Time left (sec)

--------------    -----------------   ---------------
```

The following example shows the errdisable summary information. In this example, port 6 is errdisabled for a BPDU guard violation.

```
device# show errdisable summary
Port 6 ERR_DiSABLED for bpduguard
```

# show default

Displays the system default settings of the device.

## Syntax

**show default** [ **values** ]

## Parameters

**values**

Displays default, maximum, current, and configured values for system parameters.

## Modes

Privileged EXEC mode

## Examples

The following output displays the system default settings.

```
device# show default
spanning tree disabled      fast port span disabled
auto sense port speed       port untagged               port flow control on
no username assigned        no password assigned        boot sys flash primary
system traps enabled        ntp disabled                radius disabled
rip disabled                ospf disabled               bgp disabled

when ip routing enabled :
ip irdp disabled            ip load-sharing enabled     ip proxy arp disabled
ip rarp enabled             ip bcast forward disabled
dvmrp disabled              pim/dm disabled
vrrp disabled               fsrp disabled

when rip enabled :
rip type:v2 only            rip poison rev enabled

ipx disabled                appletalk disabled
```

The following output displays the system default parameter values.

```
device# show default values
sys log buffers:50         mac age time:300 sec       telnet sessions:5

ip arp age:10 min          bootp relay max hops:4     ip ttl:64 hops
ip addr per intf:24

when multicast enabled :
igmp group memb.:260 sec   igmp query:125 sec         hardware drop: enabled

when ospf enabled :
ospf dead:40 sec           ospf hello:10 sec          ospf retrans:5 sec
ospf transit delay:1 sec

when bgp enabled :
bgp local pref.:100        bgp keep alive:60 sec      bgp hold:180 sec
bgp metric:10              bgp local as:1             bgp cluster id:0
bgp ext. distance:20       bgp int. distance:200      bgp local distance:200

System Parameters    Default    Maximum    Current    Configured
ip-arp               4000       64000      4000       4000
ip-static-arp        512        6000       512        512
ip-cache             10000      32768      10000      10000
ip-filter-port       3071       3071       3071       3071
ip-filter-sys        3072       16384      3072       3072
l3-vlan              32         1024       32         32
ip-qos-session       1024       16000      1024       1024
mac                  32768      32768      32768      32768
ip-route             12000      15168      12000      12000
ip-static-route      64         2048       64         64
vlan                 64         4095       64         64
spanning-tree        32         254        32         32
mac-filter-port      32         256        32         32
mac-filter-sys       64         512        64         64
ip-subnet-port       24         128        24         24
session-limit        8192       16384      8192       8192
view                 10         65535      10         10
virtual-interface    255        512        255        255
hw-ip-next-hop       13312      14336      13312      13312
hw-traffic-condition 50         1024       50         50
rmon-entries         1024       32768      1024       1024
igmp-snoop-mcache    512        8192       512        512
mld-snoop-mcache     512        8192       512        512
ip6-route            5120       5120       5120       5120
ip6-static-route     178        1024       1024       1024
ip6-cache            5120       5120       5120       5120
msdp-sa-cache        4096       8192       4096       4096
gre-tunnels          16         64         24         24
ip-vrf               32         32         32         32
ip-route-default-vrf 12000      15168      12000      12000
ip6-route-default-vr 5120       5120       5120       5120
ip-route-vrf         1024       15168      1024       1024
ip6-route-vrf        100        5120       100        100
pim-hw-mcache        1024       6144       1024       1024
pim6-hw-mcache       512        1024       512        512
igmp-snoop-group-add 4096       8192       4096       4096
mld-snoop-group-addr 4096       8192       4096       4096
mac-notification-buf 4000       16000      4000       4000
dot1x-mka-policy-gro 8          8          8          8
openflow-flow-entrie 3072       12288      3072       3072
openflow-pvlan-entri 40         40         40         40
openflow-unprotected 40         40         40         40
openflow-nexthop-ent 1024       3072       1024       1024
max-ip-mac           120        248        120        120
max-dhcp-snoop-entri 1024       3072       1024       1024
max-static-inspect-a 512        1024       512        512
pms-global-pool      8192       8192       8192       8192
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | The command output displayed PMS global pool default values. |

# show default values

Displays default, maximum, current, and configured values for system maximum parameters.

## Syntax

**show default values**

## Modes

Privileged EXEC mode

## Examples

This example does not show complete output; it shows only PIM hardware mcache values.

```
Device(config)#show default values
System Parameters      Default    Maximum    Current    Configured
pim-hw-mcache          1024       6144       1500       1500
```

This example does not show complete output; it shows only PIM6 hardware mcache values.

```
Device(config)#show default values
System Parameters      Default    Maximum    Current    Configured
pim6-hw-mcache         512        1024       1024       1024
```

This example does not show complete output; it shows only MLD mcache values.

```
Device(config)#show default values
System Parameters      Default    Maximum    Current    Configured
mld-snoop-mcache       512        8192       512        512
```

This example does not show complete output; it shows only IGMP group values.

```
Device(config)#show default values
System Parameters      Default    Maximum    Current    Configured
igmp-snoop-group-add   4096       8192       5000       5000
```

This example does not show complete output; it shows only MLD group values.

```
Device(config)#show default values
System Parameters      Default    Maximum    Current    Configured
MLD-snoop-group-addr   4096       8192       5000       5000
```

# show dlb-internal-trunk-hash

Displays the dynamic load balancing (DLB) hashing method for inter-packet-processor (inter-pp) links that connect master and slave units in ICX 7450-48 devices.

## Syntax

**show dlb-internal-trunk-hash**

## Modes

Global configuration mode

## Examples

The following example displays the hashing method in effect for inter-pp links on an ICX 7450-48 device.

```
ICX7450-48P Router(config)#show dlb-internal-trunk-hash
Internal trunk mode: spray-mode
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show dot1x

Displays information about the 802.1X configuration.

## Syntax

**show dot1x**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

dot1x configuration mode

## Command Output

The **show dot1x** command displays the following information:

| Output field | Description |
|---|---|
| PAE Capability | The Port Access Entity (PAE) role for the device. This is always "Authenticator Only". |
| system-auth-control | Whether system authentication control is enabled on the device. The **dot1x-enable** command enables system authentication control on the device. |
| re-authentication | Whether periodic re-authentication is enabled on the device. When periodic re-authentication is enabled, the device automatically re-authenticates clients every 3,600 seconds by default. |
| global-filter-strict-security | Whether strict security mode is enabled or disabled globally. |
| quiet-period | When the device is unable to authenticate a client, the amount of time the device waits before trying again (default 60 seconds). |
| tx-period | When a client does not send back an EAP-response/identity frame, the amount of time the device waits before retransmitting the EAP-request/identity frame to a client (default 30 seconds). |
| supptimeout | When a client does not respond to an EAP-request frame, the amount of time before the device retransmits the frame. |
| servertimeout | When the Authentication Server does not respond to a message sent from the client, the amount of time before the device retransmits the message. |
| maxreq | The number of times the device retransmits an EAP-request/identity frame if it does not receive an EAP-response/identity frame from a client (default 2 times). |
| reAuthMax | The maximum number of re-authentication attempts. |
| re-authperiod | How often the device automatically re-authenticates clients when periodic re-authentication is enabled (default 3,600 seconds). |
| Protocol Version | The version of the 802.1X protocol in use on the device. |

# Examples

The following example displays information about the 802.1X configuration.

```
device# show dot1x

PAE Capability                    : Authenticator Only
system-auth-control               : Enable
re-authentication                 : Disable
global-filter-strict-security     : Enable
quiet-period                      : 60 Seconds
tx-period                         : 30 Seconds
supptimeout                       : 30 Seconds
servertimeout                     : 30 Seconds
maxreq                            : 2
reAuthMax                         : 2
re-authperiod                     : 3600 Seconds
Protocol Version                  : 1
```

# show dot1x configuration

Displays detailed information about the 802.1X configuration.

## Syntax

**show dot1x configuration**[ **all** | **stack-unit** *id* | **ethernet** *unit*/*slot*/*port* ]

## Parameters

**all**

Displays information about the 802.1X configuration on all ports.

**ethernet** *unit*/*slot*/*port*

Displays information about the 802.1X configuration on a specific port.

**stack-unit** *id*

Displays 802.1X configuration for the specified stack unit.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

dot1x configuration mode

## Command Output

The **show dot1x configuration** command displays the following information:

| Output field | Description |
|---|---|
| PAE Capability | The Port Access Entity (PAE) role for the Ruckus device. This is always "Authenticator Only". |
| system-auth-control | Whether system authentication control is enabled on the device. The **dot1x-enable** command enables system authentication control on the device. |
| Number of Ports enabled | The number of ports on which 802.1X authentication is enabled. |
| Re-authentication | Whether periodic re-authentication is enabled on the device. When periodic re-authentication is enabled, the device automatically re-authenticates clients every 3,600 seconds by default. |
| Authentication-fail-action | The configured authentication-failure action. This can be Restricted VLAN or Block Traffic. |
| Mac Session Aging | Whether aging for dot1x-MAC-sessions has been enabled or disabled for permitted or denied dot1x-MAC-sessions. |
| Mac Session max-age | The configured software aging time for dot1x-MAC-sessions. |
| Protocol Version | The version of the 802.1X protocol in use on the device. |
| quiet-period | When the device is unable to authenticate a client, the amount of time the device waits before trying again (default 60 seconds). |

| Output field | Description |
|---|---|
| tx-period | When a client does not send back an EAP-response/identity frame, the amount of time the device waits before retransmitting the EAP-request/identity frame to a client (default 30 seconds). |
| supptimeout | When a client does not respond to an EAP-request frame, the amount of time before the device retransmits the frame. |
| servertimeout | When the Authentication Server does not respond to a message sent from the client, the amount of time before the device retransmits the message. |
| maxreq | The number of times the device retransmits an EAP-request/identity frame if it does not receive an EAP-response/identity frame from a client (default 2 times). |
| reAuthmax | The maximum number of re-authentication attempts. |
| re-authperiod | How often the device automatically re-authenticates clients when periodic re-authentication is enabled (default 3,600 seconds). |
| global strict security | Whether strict security mode is enabled or disabled globally. |

The **show dot1x configuration ethernet** *slot/port* command displays the following information:

| Output field | Description |
|---|---|
| Port-Control | The configured port control type for the interface. This can be one of the following types:<br>• force-authorized: The controlled port is placed unconditionally in the authorized state, allowing all traffic. This is the default state for ports on the device.<br>• force-unauthorized: The controlled port is placed unconditionally in the unauthorized state. No authentication takes place for any connected 802.1X clients.<br>• auto - The authentication status for each 802.1X client depends on the authentication status returned from the RADIUS server. |
| filter strict security | Whether strict security mode is enabled or disabled on the interface. |
| Action on RADIUS timeout | The action taken for the client MAC session on this port upon a RADIUS timeout. |
| Authentication-fail-action | The configured authentication-failure action. This can be Restricted VLAN or Block Traffic. |
| PVID State | The port default VLAN ID (PVID) and the state of the port PVID. The PVID state can be one of the following:<br>• Normal - The port PVID is not set by a RADIUS server, nor is it the restricted VLAN.<br>• RADIUS - The port PVID was dynamically assigned by a RADIUS server.<br>• RESTRICTED - The port PVID is the restricted VLAN. |
| Original PVID | The originally configured (not dynamically assigned) PVID for the port. |
| Authorized PVID ref count | The number of authenticated MAC sessions on this port's current PVID (port default VLAN ID). |
| Restricted PVID ref count | The number of MAC sessions on the port that failed authentication and are now in the restricted VLAN (which should be the port's current PVID). |
| Radius assign PVID ref count | The number of times the port has changed PVIDs due to RADIUS VLAN assignment. |
| num mac sessions | The number of dot1x-MAC-sessions on the port. |
| num mac authorized | The number of authorized dot1x-MAC-sessions on the port. |
| num Dynamic Tagged Vlan | The number of dynamically tagged VLANs on the port. |
| Number of Auth filter | The number of dynamic MAC filters applied to the port. |

# Examples

The following example displays information about the 802.1X configuration.

```
device# show dot1x configuration
PAE Capability                : Authenticator Only
system-auth-control           : Enable
Number of Ports enabled       : 3
Re-Authentication             : Disabled
Authentication-fail-action    : Per Port
Mac Session Aging             : Enabled
Mac Session max-age           : 120 seconds
Protocol Version              : 1
quiet-period                  : 60 Seconds
tx-period                     : 30 Seconds
supptimeout                   : 30 Seconds
servertimeout                 : 30 Seconds
maxreq                        : 2
reAuthmax                     : 2
re-authperiod                 : 3600 Seconds
global strict security        : Enable
```

The following example displays information about the 802.1X configuration on an individual port.

```
device# show dot1x configuration ethernet 4/1/12
Port-Control                  : control-auto
filter strict security        : Enable
Action on RADIUS timeout      : Restart authentication
Authentication-fail-action    : Restricted VLAN(299)
PVID State                    : Normal (1)
Original PVID                 : 1
Authorized PVID ref count     : 2
Restricted PVID ref count     : 0
Radius assign PVID ref count  : 0
num mac sessions              : 2
num mac authorized            : 2
num Dynamic Tagged Vlan       : 0
Number of Auth filter         : 0
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | The command was modified to include the stack-id option. |

# show dot1x ip-acl

Displays the Layer 3 ACLs for 802.1X authentication.

## Syntax

**show dot1x ip-acl** { **all** | **stack-unit** *id* | **ethernet** *unit*/*slot*/*port* }

## Parameters

**all**

Specifies the ACLs at the global level.

**ethernet** *unit*/*slot*/*port*

Specifies the ACLs at the interface level.

**stack-unit** *id*

Displays 802.1X authentication ACLs for the specified stack unit.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show dot1x ip-acl** command displays the following information.

| Output field | Description |
|---|---|
| Port | The port number. |
| MAC Addr | The MAC address of the client. |
| Inbound IPv4 ACL | The IPv4 ACL applied to the authenticated port in the inbound direction. |
| Outbound IPv4 ACL | The IPv4 ACL applied to the authenticated port in the outbound direction. |
| Inbound IPv6 ACL | The IPv6 ACL applied to the authenticated port in the inbound direction. |
| Outbound IPv6 ACL | The IPv6 ACL applied to the authenticated port in the inbound direction. |

## Examples

The following example displays 802.1X IP ACL authentication information for Ethernet interface 1/1/15.

```
device# show dot1x ip-acl ethernet 1/1/15
-----------------------------------------------------------------
Port      MAC              Inbound     Outbound   Inbound   Outbound
          Addr             IPv4 ACL    IPv4 ACL   IPv6 ACL  IPv6 ACL
-----------------------------------------------------------------
1/1/15     0180.c200.0003   10          11          20        21
1/1/15     0180.c300.0005   100         101         120       121
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | The command output was updated. |
| 08.0.70 | The command was modified to include the stack-id option. |

# show dot1x mac-address-filter

Displays the MAC address filters active on the device.

## Syntax

**show dot1x mac-address-filter** [ **all** | **ethernet** *unit/slot/port* | **user-defined** ]

## Parameters

**all**

Displays dynamically applied MAC address filters active on the device.

**ethernet** *unit/slot/port*

Displays dynamically applied MAC address filters active on an interface.

**user-defined**

Displays user-defined MAC address filters active on the device.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

dot1x configuration mode

## Examples

The following example displays dynamically applied MAC address filters active on an interface.

```
device# show dot1x mac-address-filter ethernet 1/1/3
Port 1/3 MAC Address Filter information:
802.1X Dynamic MAC Address Filter :
mac filter-group 2
Port default MAC Address Filter:
No mac address filter is set
```

# show dot1x mac-filter

Shows the layer 2 ACLs for 802.1X authentication.

## Syntax

**show dot1x mac-filter** { **all** | **ethernet** *device*/*slot*/*port* }

## Parameters

**all**

Specifies the ACLs at the global level.

**ethernet** *device*/*slot*/*port*

Specifies the ACLs at the interface level.

## Modes

Global configuration

Interface configuration

## Usage Guidelines

## Command Output

The **show mac-filter** command displays the following information:

| Output field | Description |
|---|---|
| Dynamic MAC filter-list | The MAC filter defined on the device. |

## Examples

The **show dot1x mac-filter** command displays the following information

```
device# show dot1x mac-filter all
802.1x MAC Address Filter information:
Port 1/1/48:
Dynamic MAC filter-list: 1
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show dot1x mac-session

Displays information about the dot1x-MAC-session on each port on the device.

## Syntax

**show dot1x mac-sessions** [ **brief** | **ip-addr** ]

## Parameters

**brief**

Displays information about the dot1x-MAC-sessions in brief.

**ip-addr**

Displays dot1x-mac-session information with an IP address instead of a MAC address.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

dot1x configuration mode

## Command Output

The **show dot1x mac-sessions** command displays the following information:

| Output field | Description |
|---|---|
| Port | The port on which the dot1x-MAC-session exists. |
| MAC/IP (username) | The MAC address of the client and the username used for RADIUS authentication. |
| Vlan | The VLAN to which the port is currently assigned. |
| Auth-State | The authentication state of the dot1x-MAC-session. This can be one of the following states:<br>• permit - The client has been successfully authenticated, and traffic from the client is being forwarded normally.<br>• blocked - Authentication failed for the client, and traffic from the client is being dropped in hardware.<br>• restricted - Authentication failed for the client, but traffic from the client is allowed in the restricted VLAN only.<br>• init - The client is in is in the process of 802.1X authentication, or has not started the authentication process. |
| Age | The software age of the dot1x-MAC-session. |
| PAE State | The current status of the Authenticator PAE state machine. This state can be INITIALIZE, DISCONNECTED, CONNECTING, AUTHENTICATING, AUTHENTICATED, ABORTING, HELD,FORCE_AUTH, or FORCE_UNAUTH. |

| Output field | Description |
|---|---|
| | **NOTE**<br>When the Authenticator PAE state machine is in the AUTHENTICATING state, if the reAuthenticate, eapStart, eapLogoff, or authTimeout parameters are set to TRUE, it may place the Authenticator PAE state machine indefinitely in the ABORTING state. If this should happen, use the **dot1x initialize** command to initialize 802.1X authentication on the port, or unplug the client or hub connected to the port, then reconnect it. |

The **show dot1x mac-session brief** command displays the following information:

| Output field | Description |
|---|---|
| Port | Information about the users connected to each port. |
| Number of users | The number of users connected to the port. |
| Number of Authorized users | The number of users connected to the port that have been successfully authenticated. |
| Dynamic VLAN | Whether the port is a member of a RADIUS-specified VLAN. |
| Dynamic ACL | Whether RADIUS-specified IP ACLs are applied to the port. |
| Dynamic MAC-Filter | Whether RADIUS-specified MAC address filters are applied to the port. |

# Examples

The following example displays information about the dot1x-MAC-session on each port on the device.

```
device# show dot1x mac-session
Port MAC/IP(username)           Vlan       Auth      ACL      Age    PAE
State State
-------------------------------------------------------------------------------
4/1/12 0044.0002.0002 :user1    10         permit    none     Ena    AUTHENTICATED
4/1/12 0044.0002.0003 :user2    10         permit    none     Ena    AUTHENTICATED
```

The following example displays information about the dot1x-MAC-session in brief.

```
device# show dot1x mac-session brief
Port           Number of    Number of           Dynamic     Dynamic   Dynamic
               users        Authorized users    VLAN        ACL       MAC-Filt
-----------------------------------------------------------------------
4/1/12         2            2                    no          no        no
```

# show dot1x sessions

Displays 802.1X authentication sessions at the global and interface levels.

## Syntax

**show dot1x sessions** { **all** | **brief** | **stack-unit** *id* | **ethernet** *unit/slot/port* }

## Parameters

**all**
Displays of 802.1X authentication sessions for all ports.

**brief**
Displays summary information for 802.1X authentication sessions.

**ethernet** *unit/slot/port*
Displays 802.1X authentication sessions for a specified Ethernet interface.

**stack-unit** *id*
Displays of 802.1X authentication sessions for the specified stack unit.

## Modes

Privileged EXEC mode

## Usage Guidelines

A client session can have an IPv4 address and multiple IPv6 addresses. When multiple addresses exist, the **show dot1x sessions** command displays all addresses for the session.

## Command Output

The **show dot1x sessions** command displays the following information.

| Output field | Description |
|---|---|
| Port | Port number. |
| MAC Addr | MAC address of the client. |
| IP Addr | IP address or addresses of the client (a session can have an IPv4 address and multiple IPv6 addresses). IP addresses of the authenticated host are only displayed when an IP ACL is applied to the interface based on the RADIUS server response. |
| User Name | User name. |
| Vlan | VLAN ID. |
| Auth State | Authentication state. |
| ACL | Specific applied ACL. |
| Session Time | Session time. |
| Age | Age of the session. |
| PAE State | Port access entity state. |

# Examples

The following example displays 802.1X sessions for all interfaces.

```
device(config)# show dot1x sessions all
-------------------------------------------------------------------------------------------------------------
-
Port    MAC             IP(v4/v6)            User    VLAN  Auth    ACL  Session Age PAE
        Addr            Addr                 Name          State       Time        State
-------------------------------------------------------------------------------------------------------------
-
2/1/25  00aa.aaaa.0000  fe80::2aa:aaff:feaa: VDI_1   130   permit  Yes  210     Ena AUTHENTICATED
                        2000::2
                        2000::4
2/1/25  00aa.aaaa.0001  fe80::2aa:aaff:feaa: VDI_2   130   permit  Yes  210     Ena AUTHENTICATED
                        3000::2
                        3000::2
```

The following example displays 802.1X authentication sessions for a specific interface.

```
device(config)# show dot1x sessions ethernet 2/1/1
-------------------------------------------------------------------------------------------
Port   MAC             IP          User    Vlan  Auth    ACL    Session   Age   PAE
       Addr            Addr        Name          State          Time            State
-------------------------------------------------------------------------------------------
2/1/1  0010.9400.1303  192.85.1.2   User1   200   permit  Yes    100       Ena   AUTHENTICATED
```

The following example displays 802.1X authentication sessions in brief.

```
device# show dot1x sessions brief
-------------------------------------------------------------------------
Port   Number of  Number of   Number of   Untagged          Dynamic   Dynamic
       Attempted  Authorized  Denied      VLAN Type         Port ACL  MAC-Filt
       Users      Users       Users
-------------------------------------------------------------------------
1/1/2  1          1           0           Radius-VLAN       No        No
1/1/3  0          0           0           Auth-Default-VLAN  No       No
1/1/4  0          0           0           Auth-Default-VLAN  No       No
1/1/5  0          0           0           Auth-Default-VLAN  No       No
2/1/1  0          0           0           Auth-Default-VLAN  No       No
2/1/2  0          0           0           Auth-Default-VLAN  No       No
2/1/4  0          0           0           Auth-Default-VLAN  No       No
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | The command output was updated. |
| 08.0.61 | The command output was modified to display multiple IPv6 addresses for a session. |
| 08.0.70 | The command was modified to include the stack-id option. |

# show dot1x sessions detail

Displays 802.1X authentication sessions at the global and interface levels.

## Syntax

**show dot1x sessions detail** { **ethernet** *unit/slot/port* }

## Parameters

**ethernet** *unit/slot/port*

Displays 802.1X authentication sessions for a specified Ethernet interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

A client session can have an IPv4 address and multiple IPv6 addresses. When multiple addresses exist, the **show dot1x sessions detail** command displays all addresses for the session.

## Command Output

The **show dot1x sessions detail** command displays the following information.

| Output field | Description |
|---|---|
| Port | Port number. |
| MAC Addr | MAC address of the client. |
| IP Addr | IP address or addresses of the client (a session can have an IPv4 address and multiple IPv6 addresses). IP addresses of the authenticated host are only displayed when an IP ACL is applied to the interface based on the RADIUS server response. |
| User Name | User name. |
| Vlan | VLAN ID. |
| Auth State | Authentication state. |
| ACL | Specific applied ACL. |
| Session Time | Session time. |
| Age | Age of the session. |
| PAE State | Port access entity state. |

# Examples

The following example displays details for 802.1X authentication sessions on a specific interface.

```
device(config)# show dot1x sessions ethernet 2/1/1
-------------------------------------------------------------------------------------------
Port   MAC             IP          User    Vlan  Auth    ACL    Session   Age   PAE
       Addr            Addr        Name          State          Time            State
-------------------------------------------------------------------------------------------
2/1/1  0010.9400.1303  192.85.1.2   User1  200   permit  Yes    100       Ena   AUTHENTICATED
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show dot1x statistics

Displays the 802.1X authentication statistics.

## Syntax

**show dot1x statistics** { **all** | **stack-unit** *id* | **ethernet** *device**/**slot**/**port* }

## Parameters

**all**

Displays the 802.1X authentication statistics for all interfaces.

**ethernet** *device**/**slot**/**port*

Displays the 802.1X authentication statistics for the specified interface.

**stack-unit** *id*

Displays 802.1X authentication statistics for the specified stack unit.

## Modes

Privileged EXEC mode

Global configuration

Interface configuration

Authentication configuration mode

## Usage Guidelines

## Command Output

The **show dot1x statistics** command displays the following information:

| Output field | Description |
|---|---|
| RX EAPOL Start | The number of EAPOL-Start frames received on the port. |
| RX EAPOL Logoff | The number of EAPOL-Logoff frames received on the port. |
| RX EAPOL Invalid | The number of invalid EAPOL frames received on the port. |
| RX EAPOL Total | The total number of EAPOL frames received on the port. |
| RX EAP Resp/Id | The number of EAP-Response/Identity frames received on the port |
| RX EAP Resp other than Resp/Id | The total number of EAP-Response frames received on the port that were not EAP-Response/Identity frames. |
| RX EAP Length Error | The number of EAP frames received on the port that have an invalid packet body length. |
| Last EAPOL Version | The version number of the last EAPOL frame received on the port. |
| Last EAPOL Source | The source MAC address in the last EAPOL frame received on the port. |
| TX EAPOL Total | The total number of EAPOL frames transmitted on the port. |
| TX EAP Req/Id | The number of EAP-Request/Identity frames transmitted on the port. |

| Output field | Description |
|---|---|
| TX EAP Req other than Req/Id | The number of EAP-Request frames transmitted on the port that were not EAP-Request/ Identity frames. |

# Examples

The following example displays 802.1X authentication statistics for port 10/2/1.

```
device# show dot1x statistics ethernet 10/2/1

Port 10/2/1 Statistics:
RX EAPOL Start : 2
RX EAPOL Logoff : 2
RX EAPOL Invalid : 0
RX EAPOL Total : 12
RX EAP Resp/Id : 4
RX EAP Resp other than Resp/Id : 4
RX EAP Length Error : 0
Last EAPOL Version : 1
Last EAPOL Source : 0022.0002.0002
TX EAPOL Total : 0
TX EAP Req/Id : 10417
TX EAP Req other than Req/Id : 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.70 | The command was modified to include the stack-id option. |

# show dot1x-mka config

Shows the MACsec Key Agreement (MKA) configuration for the device.

## Syntax

**show dot1x-mka config**

## Modes

EXEC, privileged EXEC, global configuration, or dot1x-mka interface mode.

## Command Output

The **show dot1x-mka config** command displays the following information:

| Output field | Description |
|---|---|
| dot1x-mka-enable | MACsec is enabled on the device. |
| enable-mka ethernet *device/slot/port* | The ethernet interfaces specified are enabled for MACsec. |
| mka-cfg-group *group-name* | The configuration details that follow are for the named MACsec MKA group. |
| key-server-priority *value* | The key server priority for MACsec transmissions on the named group is set at this value. |
| macsec cipher-suite gcm-aes-128<br>or<br>macsec cipher-suite gcm-aes-128 integrity-only | MACsec encryptions between members of the group are encrypted.<br>or<br>ICV checking only is performed, but no encryption is performed. |
| macsec confidentiality-offset *value* | The byte offset used for encrypted data is set to the value shown. Allowable values are 0, 30 (the first 30 bytes of data are not encrypted), and 50 (the first 50 bytes of data are not encrypted). |
| macsec frame-validation { check \| discard } | For transmissions between MKA group members, indicates whether the MACsec frame header is checked and what action is taken for invalid frames (counted or discarded). |
| macsec-replay protection { strict \| out-of-order window-size *value* } | Replay protection is enabled. The type of protection is shown as strict (discard any frame received out of sequence) or as allowing receipt of out-of-sequence frames within the specified window. |
| key *value* name *value* | The pre-shared key is set to this value and name for the MKA configuration group. Both key and name are hexadecimal strings. |
| enable ethernet *device/slot/port*<br>mka-cfg-group *name*<br>key *hexadecimal value* name *hexadecimal value* | The specified interface is enabled for MACsec. The interface belongs to the named MKA group, and the interface uses the pre-shared key shown to confirm peers with which it can communicate. |

# Examples

The following example displays MACsec configuration information for a device with MACsec enabled. Two MKA groups, test1 and group1, are configured. Interfaces with either group of parameters applied could form secure channels because the groups have the same pre-shared key.

```
device(config-dot1x-mka-1/3/2)# show dot1x-mka config

dot1x-mka-enable
 mka-cfg-group test1
  key-server-priority 5
  macsec cipher-suite gcm-aes-128 integrity-only
  macsec confidentiality-offset 30
  macsec frame-validation strict
mka-cfg-group group1
  key-server-priority 20
  macsec cipher-suite gcm-aes-128
  macsec confidentiality-offset 30
 enable-mka ethernet 1/3/2
  mka-group test1
  pre-shared-key 135bd758 b0ee5c11 c55ff6ab 19fdb199 key-name 96437a93 ccf10d9d fe347846 cce52c7d
 enable-mka ethernet 1/3/3
  mka-group group1
  pre-shared-key 135bd758 b0ee5c11 c55ff6ab 19fdb199 key-name 96437a93 ccf10d9d fe347846 cce52c7d
 enable-mka ethernet 1/3/4
  mka-group group1
  pre-shared-key 135bd758 b0ee5c11 c55ff6ab 19fdb199 key-name 96437a93 ccf10d9d fe347846 cce52c7d
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# show dot1x-mka config-group

Shows details for the specified MACsec Key Agreement (MKA) groups configured on this device, or for a designated MKA group.

## Syntax

**show dot1x-mka config-group** *group-name*

## Parameters

***group-name*** Limits the group configuration displayed to the named MKA group.

## Modes

EXEC, privileged EXEC, global configuration, or dot1x-mka interface mode.

## Command Output

The **show dot1x-mka config-group** command displays the following information:

| Output field | Description |
|---|---|
| mka-cfg-group | The configuration details that follow are for the specified MACsec MKA group. |
| key-server-priority | The key-server priority for MACsec transmissions on the named group is set at te specified value. |
| macsec cipher-suite gcm-aes-128<br>or<br>macsec cipher-suite gcm-aes-128 integrity-only | MACsec transmissions are encrypted.<br>or<br>ICV checking only is performed. |
| macsec confidentiality-offset | The byte offset used for encrypted data is set to the value shown. Allowable values are 0, 30 (the first 30 bytes of data are not encrypted), and 50 (the first 50 bytes of data are not encrypted). |
| macsec frame-validation {check \| discard} | Indicates whether the MACsec frame header is checked and what action is taken for invalid frames (counted or discarded). |
| macsec replay-protection {strict \| out-of-order window-size *size*} | Replay protection is enabled. The type of protection is shown as strict (discard any frame received out of sequence) or as allowing receipt of out-of-sequence frames within the specified window. |

## Examples

The following example lists the configuration details for MKA group test1.

```
device(config-dot1x-mka-1/3/2)# show dot1x-mka config-group test1

mka-cfg-group test1
  key-server-priority 5
  macsec cipher-suite gcm-aes-128 integrity-only
  macsec confidentiality-offset 30
  macsec frame-validation check
        macsec replay-protection strict
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# show dot1x-mka sessions

Displays a summary of all MACsec Key Agreement (MKA) sessions on the device.

## Syntax

**show dot1x-mka sessions brief**

**show dot1x-mka sessions ethernet** *device/slot/port*

## Parameters

| | |
|---|---|
| **brief** | Displays a brief status of all MKA sessions. |
| **ethernet *device/slot/port*** | Displays MKA sessions that are active on a specified Ethernet interface. The Ethernet interface is specified by device position in stack, slot on the device, and interface on the slot. |

## Modes

EXEC, Privileged EXEC, global configuration, or dot1x-mka interface mode.

## Command Output

The **show dot1x-mka sessions** command with the **brief** option displays the following information:

| Output field | Description |
|---|---|
| Port | Designates the interface for which MACsec information is listed (by device, slot, and port). |
| Link-Status | Indicates whether the link is up or down. |
| MKA-Status | Indicates whether a secure channel has been established. |
| Key-Server | Indicates whether the interface is operating as a key-server. |
| Negotiated Capability | Indicates MACsec parameters configured on the designated interface. |

The **show dot1x-mka sessions** command with the **ethernet** interface options displays the following information:

| Output field | Description |
|---|---|
| Interface | The information that follows applies to the designated interface. |
| MKA cfg group Name | The designated MKA configuration group has been applied to the designated interface. |
| DOT1X-MKA Enabled (Yes, No) | Indicates whether MACsec is enabled for the designated interface. |
| DOT1X-MKA Active (Yes, No) | Indicates whether MACsec is active on the interface. |
| Key Server (Yes, No) | Indicates whether the MACsec key-server is active over the interface. |
| Configuration Status: | The following fields describe the MKA configuration applied to the interface. |
| Enabled (Yes, No) | Indicates whether MACsec is currently enabled. |
| Capability (Integrity and or confidentiality) | Indicates whether ICV checks are being performed on MACsec frames and whether encryption is being applied. |
| Desired (Yes, No) | Indicates whether port is interested in becoming the key-server. |
| Protection (Yes, No) | Indicates whether replay protection is applied to the interface. |
| Frame Validation (Yes, No) | Indicates whether frames received are being checked for valid MACsec headers. |

| Output field | Description |
|---|---|
| Replay Protection (Strict, Out of Order) | Indicates that replay protection is configured and whether frames must be received in exact order or within an allowable window. |
| Replay Protection Size | Indicates the allowable window size within which frames may be received. |
| Cipher Suite (GCM-AES-128) | Specifies the cipher suite used for ICV checking, encryption, and decryption. |
| Key Server Priority (1 to 127) | Specifies the key-server priority configured on the interface. |
| Secure Channel Information | The following fields describe a secure channel established on this interface. |
| Local SCI | Provides the hexadecimal value of the Secure Channel Identifier for this channel. |
| Member Identifier | Provides the MACsec number assigned to the MKA peer. |
| Message Number | Provides the Message Number contained in Hello packets from this MKA peer. Hello packets are exchanged to determine peer status, MACsec capabilities, and SAK Key Identifier. |
| Latest SAK Status (RX and or TX) | Indicates the Secure Association Key (SAK) state. |
| Latest SAK AN | Provides the Association Number for the most recently active Secure Association Key. |
| Latest SAK KI | Provides the Key Identifier for the most recently active Secure Association Key. |
| Negotiated Capability (Integrity and or Confidentiality with offset) | Indicates whether ICV checking, encryption, and a confidentiality offset have been applied on the secure channel. (The negotiated capability may differ from parameters configured on the interface when it does not have key-server status.) |
| Peer Information: | The output fields that follow provide information on actual and potential MACsec peer interfaces. |
| State (Live or Potential) | Indicates whether the peer is considered a live peer or a potential peer for MKA protocol. |
| Member Identifier | Designates the peer by its Member Identifier, a hexadecimal value. |
| Message Number | Provides the Message Number that appears in Hello packets from the designated peer interface as a hexadecimal value. |
| SCI | Provides the peer's Secure Channel Identifier. |
| Priority | Provides the key-server priority configured on the peer interface. |

# Examples

In the following example, all enabled MKA interfaces on the device are listed, along with configured parameters and current status.

```
device(config-dot1x-mka-1/3/2)# show dot1x-mka sessions brief

Port    Link-Status  MKA-Status  Key-Server  Negotiated Capability

1/3/2   Down         Pending     ---         ---
1/3/3   Up           Secured     No          Integrity, Confidentiality with Off. 30
1/3/4   Up           Secured     No          Integrity, Confidentiality with Off. 30
```

The following example lists MKA sessions that are active on Ethernet interface 1/3/3 (device 1, slot 3, port 3), with configuration details for each active interface.

```
device(config-dot1x-mka-1/3/3)# show dot1x-mka sessions ethernet 1/3/3

Interface                : 1/3/3

  MACsec Status          : Secured
  DOT1X-MKA Enabled      : Yes
  DOT1X-MKA Active       : Yes
  Key Server             : No

Configuration Status:
  Enabled                : Yes
  Capability             :  Integrity,  Confidentiality
  Desired                : Yes
  Protection             : Yes
  Frame Validation       : Disable
  Replay Protection      : Strict
  Replay Protection Size : 0
  Cipher Suite           : GCM-AES-128
  Key Server Priority    : 20

  Local SCI              : 748ef8344a510082
  Member Identifier      : 802ed0536fcafc43407ba222
  Message Number         : 8612

Secure Channel Information:
  Latest SAK Status      : Rx & Tx
  Latest SAK AN          : 0
  Latest KI              : d08483062aa9457e7c2470e300000001
  Negotiated Capability  : Integrity, Confidentiality with offset 30

Peer Information:
State        Member Identifier           Message Number    SCI                Priority
-----        -----------------           --------------    ----------------   --------
Live         d08483062aa9457e7c2470e3              8527    748ef83443910082         20
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# show dot1x-mka statistics

Displays current MACsec Key Agreement (MKA) statistics on the interface.

## Syntax

**show dot1x-mka statistics ethernet** *unit*/*slot*/*port*

## Parameters

**ethernet** *unit*/*slot*/*port*

Ethernet interface for which MKA statistics are to be displayed. The unit number is 1 for a standalone unit or the stack ID for a stack member.

## Modes

EXEC, privileged EXEC, global configuration, or dot1x-mka interface mode.

## Usage Guidelines

It is recommended that you use the **clear dot1x-mka statistics** command to clear results of the previous **show dot1x-mka statistics** command before re-executing it.

## Command Output

The **show dot1x-mka statistics** command displays the following information:

| Output field | Description |
|---|---|
| Interface (device/slot/port) | The output fields describe MACsec activity for the designated interface. |
| MKA in Pkts | MKA protocol packets received |
| MKA in SAK Pkts | MKA protocol packets received containing a SAK |
| MKA in Bad Pkts | MKA protocol packets received that are bad |
| MKA in Bad ICV Pkts | MKA protocol packets received with a bad ICV |
| MKA in Mismatch Pkts | MKA protocol packets received with mismatched CAK |
| MKA out Pkts | MKA protocol packets transmitted |
| MKA out SAK Pkts | MKA protocol packets transmitted containing a SAK |
| Number of SAK | Total number of SAKs received |

# Examples

The following example shows MKA statistics for Ethernet interface 1/3/3 (device 1, slot 3, port 3), which is transmitting and receiving MACsec frames.

```
device(config-dot1x-mka-1/3/3)# clear dot1x-mka statistics ethernet 1/3/3
device(config-dot1x-mka-1/3/3)# show dot1x-mka statistics ethernet 1/3/3

Interface                : 1/3/3

MKA in Pkts              : 8585
MKA in SAK Pkts          : 1
MKA in Bad Pkts          : 0
MKA in Bad ICV Pkts      : 0
MKA in Mismatch Pkts     : 0
MKA out Pkts             : 8687
MKA out SAK Pkts         : 0
Number of SAK            : 1
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# show eee-statistics

Displays the global energy efficient statistics.

## Syntax

**show eee-statistics**

## Modes

Global configuration mode

## Usage Guidelines

## Command Output

The **show eee-statistics** command displays the following information:

| Output field | Description |
|---|---|
| Port | The port number. |
| EEE-State | Displays if Energy Efficient Ethernet is enabled or disabled. If disabled then all the counters will be 0. If EEE is enabled, then these counters will be updated. |
| TXEventCount | TX EEE Low Power Idle (LPI) event counter. This counter specifies the number of times the LPI mode has been enforced by EEE on Transmit side. |
| TXDuration | TX EEE LPI duration counter. This is an LPI event duration counter on the transmit path which gets updated if the port is in LPI mode. |
| RXEventCount | RX EEE LPI event counter. This counter specifies the number of times the LPI mode has been enforced by EEE on the receive side. |
| RXDuration | RX EEE LPI duration counter. This is an LPI event duration counter on the receive path which gets updated if the port is in LPI mode. |

# Examples

The following example displays Energy Efficient Ethernet globally.

```
device# show eee-statistics
Port        EEE-State  TXEventCount       TXDuration        RXEventCount                RXDuration
1/1/1       Enable       0                    0                 0                          0
1/1/2       Enable       0                    0                 0                          0
1/1/3       Enable       17                 2551234             16                       2561886
1/1/4       Enable       17                 2545628             16                       50953524
1/1/5       Enable       2                  2550749             2                        50952549
1/1/6       Enable       1                  2543935             1                        2551760
1/1/7       Enable       17                 2549030             17                       2550750
1/1/8       Enable       2                  419455              16                       50952710
1/1/9       Enable       1                  424565              1                        50950470
1/1/10      Enable       17                 2549030             1                        2549101
1/1/11      Enable       2                  419455              2                        424563
1/1/12      Enable       1                  424565              10                       50945833
1/1/13      Enable       2                  1526709             10                       1532337
1/1/14      Enable       10                 1531808             2                        2561886
1/1/15      Enable       10                 1531391             2                        1531834
1/1/16      Enable       2                  1526292             10                       50945548
1/1/17      Enable       2                  1542560             10                       50957135
1/1/18      Enable       10                 1537443             2                        1542565
1/1/19      Enable       10                 1528600             2                        1533722
1/1/20      Enable       2                 1533717              10                       50948350
1/1/21      Enable       2                 1533203              10                       50947920
1/1/22      Enable       10                 1528087             2                        1533230
1/1/23      Enable       10                 1527677             2                        1532799
1/1/24      Enable       2                 1532794              10                       50947596
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show eee-statistics ethernet

Displays the Energy Efficient Ethernet statistics on a specific interface.

## Syntax

**show eee-statistics ethernet** *stackid/slot/port*

## Modes

Global configuration mode

## Usage Guidelines

## Command Output

The **show eee-statistics ethernet** command displays the following information:

| Output field | Description |
|---|---|
| Port | The port number. |
| EEE-State | Displays if Energy Efficient Ethernet is enabled or disabled. If disabled then all the counters will be 0. If EEE is enabled, then these counters will be updated. |
| TXEventCount | TX EEE Low Power Idle (LPI) event counter. This counter specifies the number of times the LPI mode has been enforced by EEE on Transmit side. |
| TXDuration | The total time from the first LPI (Low Power Idle) signal transmission. This is an LPI event duration counter on the transmit path which gets updated if the port is in LPI mode. |
| RXEventCount | The LPI signal reception count. This counter specifies the number of times the LPI mode has been enforced by EEE on the receive side. |
| RXDuration | Total time from the first LPI signal reception. This is an LPI event duration counter on the receive path which gets updated if the port is in LPI mode. |

## Examples

The following example displays energy efficient statistics on a specific interface.

```
device(config)# show eee-statistics ethernet 1/1/4

Port      EEE-State       TXEventCount       TXDuration        RXEventCount        RXDuration
1/1/4      Enable           17                  2545628            16                  50953524
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show erspan

Displays the ERSPAN profiles.

## Syntax

**show erspan** [ **profile** *profile-number* ]

## Parameters

**profile**

Specifies the profile number to display.

*profile-number*

Specifies the profile number. Valid values are from 1 through 4.

## Modes

Global configuration mode

## Command Output

The **show erspan** command displays the following information:

| Output field | Description |
|---|---|
| Profile | The profile number. |
| Type | The type of profile (ERSPAN). |
| Mirror destination | Indicates whether the mirror destination is reachable or unreachable. |
| Destination IP | The IP address of the destination host. |
| Destination MAC | The MAC address of the destination host. |
| Source IP | The IP address of the source router. |
| Source MAC | The MAC address of the source router. |
| Ports monitored | The ports that are being monitored. |
| HW destination id for each device | The hardware destination ID for each device being monitored. The ID is in the form stack_id/device:dest_id. |

# Examples

The following example displays all of the ERSPAN profiles. In this example, ERSPAN mirroring has been enabled for profile 1, but has not yet been enabled for profile 2.

```
device(config)# show erspan

 Profile 1
 Type            ERSPAN
 Mirror destination Not reachable.
 Destination IP   1.1.1.1
 Destination MAC  0000.5e00.5300
 Source IP        2.2.2.2
 Source MAC       0000.5300.5312
 Outgoing port    INVALID
 Ports monitored:
  Input monitoring      : (U1/M1)   1
  Output monitoring     : (U1/M1)   1
 HW destination id for each device:
 stack_id/device:dest_id

 Profile 2
 Type            ERSPAN
 Mirror destination Not reachable.
 Destination IP   3.3.3.3
 Destination MAC  0000.5e00.5300
 Source IP        2.2.2.2
 Source MAC       0000.5300.5312
 Ports monitored:
 HW destination id for each device:
 stack_id/device:dest_id
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show ethernet loopback interfaces

Displays the status and details of each Ethernet loopback-enabled port and the associated VLANs.

## Syntax

**show ethernet loopback interfaces** [ **brief** | **port** *stackid/slot/port* | **vlan** *vlan-id* ]

## Parameters

**brief**

Displays the Ethernet loopback information in brief mode.

**port**

Displays the status and details of each port.

*stackid/slot/port*

Specifies the port number.

**vlan**

Displays the status and details of a VLAN.

*vlan-id*

Specifies the VLAN ID.

## Modes

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

## Command Output

The **show ethernet loopback interfaces** command displays the following information:

| Output field | Description |
|---|---|
| Interface Type | Type of interface (VLAN-aware or VLAN-unaware) |
| Interface Port | Interface ID (Port number) |
| Interface Mode | Flow classification mode (Flow-aware or Flow-unaware) |
| Flow Mode DA/SA | Destination and Source MAC address of the flow |

# Examples

The following example shows the output of the **show ethernet loopback interfaces** command.

```
device(config-vlan-10)# show ethernet loopback interfaces

ETHERNET LOOPBACK INTERFACE [1/1/11] (In Service)
Interface Type  : PORT
Interface Port  : 1/1/11
Interface Mode  : FLOW-UNAWARE
Flow Mode DA/SA : ANY/ANY
```

The following example shows the output of the **show ethernet loopback interfaces brief** command.

```
device(config-vlan-10)# show ethernet loopback interfaces brief
PORT         TYPE  VLANS  STATUS  OP-MODE      D-MAC           S-MAC
==========|======|=====|========|=======|==============|==============
1/1/11     |  PORT|    0|    ACTV|FLOW-U |            ANY|            ANY
1/1/12     |  VLAN|    1|    ACTV|FLOW-A |1111.2222.3333|4444.5555.5555
```

The following example shows the output of the **show ethernet loopback interfaces port** command.

```
device(config-vlan-10)# show ethernet loopback interfaces port 1/1/1
ETHERNET LOOPBACK INTERFACE [1/1/1] (In Service)
Interface Type  : PORT
Interface Port  : 1/1/1
Interface Mode  : FLOW-UNAWARE
Flow Mode DA/SA : ANY/ANY
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show ethernet loopback resources

Displays the available resources and the resources that are used by loopback testing.

## Syntax

**show ethernet loopback resources**

## Modes

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

## Command Output

The **show ethernet loopback resources** command displays the following information:

| Output field | Description |
|---|---|
| Interface Resource | Maximum number of ports that can be enabled with Ethernet loopback. |
| H/W Pool Resource | Maximum hardware resource for loopback. |

## Examples

The following example shows the output of the **show ethernet loopback resources** command.

```
device(config)# show ethernet loopback resources
Ethernet Loopback Resource:
     RESOURCE NAME       MAX      USED    AVAILABLE
======================|========|========|===========
     Interface Resource|     20|       0|      20
     H/W Pool Resource|      40|       0|      40
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show fdp entry

Displays detailed Foundry Discovery Protocol (FDP) and Cisco Discovery Protocol (CDP) information for all neighbor devices or for a specific device.

## Syntax

**show fdp entry** { ***** | *device-id* }

## Parameters

*****

Displays detailed FDP updates for all neighbor devices.

*device-id*

Specifies the device ID of the FDP neighbor entry for which the update information is to be displayed. The value is an ASCII string.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show fdp entry** command displays the following information.

| Output field | Description |
|---|---|
| Device ID | The host name of the neighbor. In addition, this field lists the VLAN memberships and other VLAN information for the neighbor port that sent the update to this device. |
| Entry address(es) | The Layer 3 protocol addresses configured on the neighbor port that sent the update to this device. If the neighbor is a Layer 2 switch, this field lists the management IP address. |
| Platform | The product platform of the neighbor. |
| Interface | The interface on which this device received the FDP or CDP update from the neighbor. |
| Port ID | The interface through which the neighbor sent the update. |
| Holdtime | The maximum number of seconds that this device can keep the information received in the update before discarding it. |
| Version | The software version running on the neighbor. |

# Examples

The following is sample output from the **show fdp entry** command.

```
device# show fdp entry FastIronB

Device ID: FastIronB configured as default VLAN1, tag-type8100
Entry address(es):
Platform: FastIron Router, Capabilities: Router
Interface: Eth 1/2/9
Port ID (outgoing port): Eth 1/2/9 is TAGGED in following VLAN(s):
9 10 11
Holdtime : 176 seconds
```

# show fdp interface

Displays Foundry Discovery Protocol (FDP) information for an interface.

## Syntax

**show fdp interface** [ **ethernet** *stack-id/slot/port* ]

## Parameters

**ethernet** *stack-id/slot/port*

Displays the FDP information for the specified Ethernet port ID.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show fdp interface** command displays the following information.

| Output field | Description |
|---|---|
| Holdtime | The maximum number of seconds that this device keeps the information received in the update before discarding it. |

## Examples

The following example shows FDP information for Ethernet port 1/2/3.

```
device# show fdp interface ethernet 1/2/3

FastEthernet1/2/3 is up, line protocol is up
Encapsulation ethernet
Sending FDP packets every 5 seconds
Holdtime is 180 seconds
```

# show fdp neighbors

Displays the Cisco neighbors about which the Ruckus ICX device has learned from Cisco Discovery Protocol (CDP) packets.

## Syntax

**show fdp neighbors** [ **detail** | **ethernet** *stack-id/slot/port* ]

## Parameters

**detail**

Displays detailed information for the Cisco neighbors.

**ethernet** *stack-id/slot/port*

Specifies the Ethernet port ID for which the information is to be displayed.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show fdp neighbors** command.

```
device# show fdp neighbors detail

Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
S - Switch, H - Host, I - IGMP, r - Repeater
(*) indicates a Cisco device
Device ID      Local Int    Holdtm   Capability  Platform     Port ID
-------------- ------------ ------   ----------  -----------  -------------
(*)Router      Eth 1/1/1      124      R          cisco RSP4   FastEthernet5/0/0
```

The following is sample output from the **show fdp neighbors detail** command.

```
device# show fdp neighbors detail

Device ID: Router
Entry address(es):
            IP address: 10.95.6.143
Platform: cisco RSP4, Capabilities: Router
Interface: Eth 1/1/1, Port ID (outgoing port): FastEthernet5/0/0
Holdtime : 150 seconds
Version :
Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-JSV-M), Version 12.0(5)T1, RELEASE SOFTWARE
(fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Thu 19-Aug-99 04:12 by xxxxxx
```

The following is sample output from the **show fdp neighbors ethernet** command.

```
device# show fdp neighbors ethernet 1/1/5

Device ID: Router
Entry address(es):
IP address: 10.95.6.143
Platform: cisco RSP4, Capabilities: Router
Interface: Eth 1/1/5, Port ID (outgoing port): FastEthernet5/0/0
Holdtime : 127 seconds
Version :
Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-JSV-M), Version 12.0(5)T1, RELEASE SOFTWARE
(fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Thu 19-Aug-99 04:12 by xxxxxx
```

# show fdp traffic

Displays packet statistics for Foundry Discovery Protocol (FDP) and Cisco Discovery Protocol (CDP).

## Syntax

**show fdp traffic**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show fdp traffic** command.

```
device# show fdp traffic

CDP/FDP counters:
Total packets output: 6, Input: 3
Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
No memory: 0, Invalid packet: 0, Fragmented: 0
Internal errors: 0
```

# show files

Displays the list of files stored in flash memory.

## Syntax

**show files** [ *dir-name* ]

## Parameters

*dir-name*

Specifies the name of a directory.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show files** command.

```
device# show files

Type    Size        Name
-----------------------
F         28203908     primary
F         27949956     secondary
F         641         startup-config.txt
F         391         stacking.boot
F         76942         debug.boot
F         638         startup-config.backup
F         0             startup-config.no

        56232476 bytes 7 File(s) in FI root

        1771020288 bytes free in FI root
        1771020288 bytes free in /
```

# show files disk0

Displays the contents of the USB flash drive.

## Syntax

**show files disk0**

## Parameters

## Modes

Enable mode

## Usage Guidelines

Insert the flash drive in the device and enter the **show files disk0** command to display the contents of the USB flash drive.

## Examples

The following example displays the contents of the USB flash drive.

```
device# show files disk0
F             681 20140611132829945ICX7450-PREM-LIC-SW.XML
F        28483780 SPS08030q066.bin
F             391 stacking.boot
F               0 sil_logs
F        28483780 pri.bin
F             391 stacking.boot1111
F            2160 running-configsp2
F            2162 startup-config.sp2
F            2160 run1
F            5344 core-file
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show flash

Displays flash memory contents on the device.

## Syntax

**show flash** [ **unit** *unit-num* ]

## Parameters

**unit** *unit-num*

Displays flash memory contents for the specified stack unit.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

Use this command to view the flash and boot images installed on the device.

The device does not have separate primary and secondary flash areas for the boot image. The flash memory module contains only one boot image.

## Command Output

The **show flash** command displays the following information.

| Output field | Description |
|---|---|
| Compressed Pri Code size | The flash code size installed in the primary flash area. |
| Compressed Sec Code size | The flash code size installed in the secondary flash area. |
| Compressed Boot-Monitor Image size | The boot code size installed in flash memory. |

## Examples

The following is sample output from the **show flash** command.

```
device# show flash
Stack unit 1:
  Compressed Pri Code size = 28893380, Version:08.0.40T211 (SPS08040b074.bin)
  Compressed Sec Code size = 28893380, Version:08.0.40T211 (SPS08040b074.bin)
  Compressed Boot-Monitor Image size = 786944, Version:10.1.05T215
  Code Flash Free Space = 1779965952
```

# show gvrp

Displays the GVRP information.

## Syntax

**show gvrp**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

GVRP configuration mode

## Command Output

The **show gvrp** command displays the following information:

| Output field | Description |
|---|---|
| Protocol state | The state of GVRP. The display shows one of the following:<br>• GVRP is disabled on the system<br>• GVRP is enabled on the system |
| GVRP BASE VLAN ID | The ID of the base VLAN used by GVRP. |
| GVRP MAX Leaveall Timer | The maximum number of milliseconds to which you can set the Leaveall timer. |
| GVRP Join Timer | The value of the Join timer. |
| GVRP Leave Timer | The value of the Leave timer. |
| GVRP Leave-all Timer | The value of the Leaveall timer. |
| Configuration that is being used | The configuration commands used to enable GVRP on individual ports. If GVRP learning or advertising is disabled on a port, this information also is displayed. |
| Spanning Tree | The type of STP enabled on the device.<br>**NOTE**<br>GVRP is only supported with Single STP. |
| Dropped Packets Count | The number of GVRP packets that the device has dropped. A GVRP packet can be dropped for either of the following reasons:<br>• GVRP packets are received on a port on which GVRP is not enabled.<br>**NOTE**<br>If GVRP support is not globally enabled, the device does not drop the GVRP packets but instead forwards them at Layer 2.<br>• GVRP packets are received with an invalid GARP protocol ID. The protocol ID must always be 0x0001. |
| Number of VLANs in the GVRP Database | The number of VLANs in the GVRP database. |

| Output field | Description |
|---|---|
| | **NOTE**<br>This number includes the default VLAN (1), the GVRP base VLAN (4093), and the Single STP VLAN (4094). These VLANs are not advertised by GVRP but are maintained as "Registration Forbidden". |
| Maximum Number of VLANs that can be present | The maximum number of VLANs that can be configured on the device. This number includes statically configured VLANs, VLANs learned through GVRP, and VLANs 1, 4093, and 4094. |

# Examples

The following example displays sample output of the **show gvrp** command.

```
device# show gvrp
GVRP is enabled on the system
GVRP BASE VLAN ID : 4093
GVRP MAX Leaveall Timer : 300000 ms
GVRP Join Timer : 200 ms
GVRP Leave Timer : 600 ms
GVRP Leave-all Timer : 10000 ms
============================================================================
Configuration that is being used:
block-learning ethe 1/1/3
block-applicant ethe 1/2/7 ethe 1/2/11
enable ethe 1/1/1 to 1/1/7 ethe 1/2/1 ethe 1/2/7 ethe 1/2/11
============================================================================
Spanning Tree: SINGLE SPANNING TREE
Dropped Packets Count: 0
============================================================================
Number of VLANs in the GVRP Database: 15
Maximum Number of VLANs that can be present: 4095
============================================================================
```

# show gvrp ethernet

Displays the GVRP information per individual port.

## Syntax

**show gvrp ethernet** *stackid/slot/port*

## Parameters

*stackid/slot/port*

Specifies the GVRP enabled ports.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

GVRP configuration mode

## Command Output

The **show gvrp ethernet** command displays the following information:

| Output field | Description |
|---|---|
| Port number | The port for which information is being displayed. |
| GVRP Enabled | Whether GVRP is enabled on the port. |
| GVRP Learning | Whether the port can learn VLAN information from GVRP. |
| GVRP Applicant | Whether the port can advertise VLAN information into GVRP. |
| Port State | The port link state, which can be UP or DOWN. |
| Forwarding | Whether the port is in the GVRP Forwarding state:<br>• NO - The port is in the Blocking state.<br>• YES - The port is in the Forwarding state. |
| VLAN Membership | The VLANs of which the port is a member. For each VLAN, the following information is shown:<br>• VLAN ID - The VLAN ID.<br>• Mode - The type of VLAN, which can be one of the following:<br>• FIXED - The port will always be a member of this VLAN and the VLAN will always be advertised on this port by GVRP. A port becomes FIXED when you configure the port as a tagged member of a statically configured VLAN.<br>• FORBIDDEN - The VLAN is one of the special VLANs that is not advertised or learned by GVRP. The following VLANs are forbidden: the default VLAN (1), the GVRP base VLAN (4093), or the Single STP VLAN (4094).<br>• NORMAL - The port became a member of this VLAN after learning about the VLAN through GVRP. The port membership in the VLAN depends on GVRP. If the VLAN is removed from the ports that send GVRP advertisements to this device, then the port will stop being a member of the VLAN. |

# Examples

The following example shows GVRP information for an individual port.

```
device# show gvrp ethernet 1/2/1
Port 1/2/1 -
 GVRP Enabled   : YES
 GVRP Learning  : ALLOWED
 GVRP Applicant : ALLOWED
 Port State     : UP
 Forwarding     : YES

 VLAN Membership:        [VLAN-ID]            [MODE]
                                1          FORBIDDEN
                                2              FIXED
                             1001             NORMAL
                             1003             NORMAL
                             1004             NORMAL
                             1007             NORMAL
                             1009             NORMAL
                             1501             NORMAL
                             2507             NORMAL
                             4001             NORMAL
                             4093          FORBIDDEN
                             4094          FORBIDDEN
```

# show gvrp statistics

Displays the GVRP statistics.

## Syntax

**show gvrp statistics** { **all** | **ethernet** *stackid/slot/port* }

## Parameters

**all**

Displays the GVRP statistics for all ports.

**ethernet** *stackid/slot/port*

Displays the GVRP statistics for a specific Ethernet port.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

GVRP configuration mode

## Command Output

The **show gvrp statistics ethernet** command displays the following information:

| Output field | Description |
|---|---|
| Leave All Received | The number of Leaveall messages received. |
| Join Empty Received | The number of Join Empty messages received. |
| Join In Received | The number of Join In messages received. |
| Leave Empty Received | The number of Leave Empty messages received. |
| Leave In Received | The number of Leave In messages received. |
| Empty Received | The number of Empty messages received. |
| Leave All Transmitted | The number of Leaveall messages sent. |
| Join Empty Transmitted | The number of Join Empty messages sent. |
| Join In Transmitted | The number of Join In messages sent. |
| Leave Empty Transmitted | The number of Leave Empty messages sent. |
| Leave In Transmitted | The number of Leave In messages sent. |
| Empty Transmitted | The number of Empty messages sent. |
| Invalid Messages/Attributes Skipped | The number of invalid messages or attributes received or skipped. This can occur in the following cases:<br>• The incoming GVRP PDU has an incorrect length.<br>• "End of PDU" was reached before the complete attribute could be parsed. |

| Output field | Description |
|---|---|
| | • The Attribute Type of the attribute that was being parsed was not the GVRP VID Attribute Type (0x01).<br>• The attribute that was being parsed had an invalid attribute length.<br>• The attribute that was being parsed had an invalid GARP event.<br>• The attribute that was being parsed had an invalid VLAN ID. The valid range is from 1 through 4095. |
| Failed Registrations | The number of failed registrations that have occurred. A failed registration can occur for the following reasons:<br>• Join requests were received on a port that was blocked from learning dynamic VLANs (GVRP Blocking state).<br>• An entry for a new GVRP VLAN could not be created in the GVRP database. |

# Examples

The following example shows the GVRP statistics for an individual port.

```
device# show gvrp statistics ethernet 1/2/1
PORT 1/2/1 Statistics:
 Leave All Received                    : 147
 Join Empty Received                   : 4193
 Join In Received                      : 599
 Leave Empty Received                  : 0
 Leave In Received                     : 0
 Empty Received                        : 588
 Leave All Transmitted                 : 157
 Join Empty Transmitted                : 1794
 Join In Transmitted                   : 598
 Leave Empty Transmitted               : 0
 Leave In Transmitted                  : 0
 Empty Transmitted                     : 1248
 Invalid Messages/Attributes Skipped   : 0
 Failed Registrations                  : 0
```

# show gvrp vlan

Displays the GVRP VLAN information.

## Syntax

**show gvrp vlan** { **all** | **brief** | *vlan-id* }

## Parameters

**all**

Displays the information for all GVRP VLANs.

**brief**

Displays the GVRP VLAN information summary.

*vlan-id*

Displays the information for a specific VLAN ID.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

GVRP configuration mode

## Command Output

The **show gvrp vlan brief** command displays the following information:

| Output field | Description |
|---|---|
| Number of VLANs in the GVRP Database | The number of VLANs in the GVRP database.<br><br>**NOTE**<br>This number includes the default VLAN (1), the GVRP base VLAN (4093), and the Single STP VLAN (4094). These VLANs are not advertised by GVRP but are included in the total count. |
| Maximum Number of VLANs that can be present | The maximum number of VLANs that can be configured on the device. This number includes statically configured VLANs, VLANs learned through GVRP, and VLANs 1, 4093, and 4094. |
| VLAN-ID | The VLAN ID. |
| MODE | The type of VLAN, which can be one of the following:<br>• STATIC - The VLAN is statically configured and cannot be removed by GVRP. This includes VLANs you have configured as well as the default VLAN (1), base GVRP VLAN (4093), and Single STP VLAN (4094).<br>• DYNAMIC - The VLAN was learned through GVRP. |
| VLAN-INDEX | A number used as an index into the internal database. |

The **show gvrp vlan** command displays the following information:

| Output field | Description |
|---|---|
| VLAN-ID | The VLAN ID. |
| VLAN-INDEX | A number used as an index into the internal database. |
| STATIC | Whether the VLAN is a statically configured VLAN. |
| DEFAULT | Whether this is the default VLAN. |
| BASE-VLAN | Whether this is the base VLAN for GVRP. |
| Timer to Delete Entry Running | Whether all ports have left the VLAN and the timer to delete the VLAN itself is running. |
| Legend | The meanings of the letter codes used in other parts of the display. |
| Forbidden Members | The ports that cannot become members of a VLAN advertised or leaned by GVRP. |
| Fixed Members | The ports that are statically configured members of the VLAN. GVRP cannot remove these ports. |
| Normal (Dynamic) Members | The ports that were added by GVRP. These ports also can be removed by GVRP. |
| MODE | The type of VLAN, which can be one of the following:<br>• STATIC - The VLAN is statically configured and cannot be removed by GVRP. This includes VLANs you have configured as well as the default VLAN (1), base GVRP VLAN (4093), and Single STP VLAN (4094).<br>• DYNAMIC - The VLAN was learned through GVRP. |

# Examples

The following example shows the output of the **show gvrp vlan brief** command.

```
device# show gvrp vlan brief
Number of VLANs in the GVRP Database: 7
Maximum Number of VLANs that can be present: 4095

        [VLAN-ID]                          [MODE]       [VLAN-INDEX]

                1                 STATIC-DEFAULT                  0
                7                         STATIC                  2
               11                         STATIC                  4
             1001                        DYNAMIC                  7
             1003                        DYNAMIC                  8
             4093          STATIC-GVRP-BASE-VLAN                  6
             4094        STATIC-SINGLE-SPAN-VLAN                  5

================================================================================
```

The following example shows the output of the **show gvrp vlan** command.

```
device# show gvrp vlan 1001
VLAN-ID: 1001, VLAN-INDEX: 7, STATIC: NO, DEFAULT: NO, BASE-VLAN: NO
Timer to Delete Entry Running: NO
Legend: [S=Slot]
Forbidden Members: None
Fixed Members: None
Normal(Dynamic) Members: (S2) 1
```

# show hardware ipv6-route

Displays the hardware information for Layer3 IPv6 hardware routes.

## Syntax

**show hardware ipv6-route** { *ipv6-address* | *ipv6-address prefix* | **device** *device-id*}

## Parameters

*ipv6-address*
Specifies an IPv6 address.

*ipv6-address prefix*
Specifies an IPv6 network number.

**device** *device_id*
Specifies the hardware device number.

## Modes

User EXEC mode

## Examples

The following example displays sample output from the **show hardware ipv6-route** command for ICX 7150 devices.

```
device> show hardware ipv6-route device 0

Total number of IPv6 hardware routes(dev:0): 6 (default:2, host:2, ip6_65-128:2)
vr: 0 fe80::/16 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0 ::/0 NH: 1025 NH hw: 101025 cmd: DROP, Outgoing vlan: 4091, L3 intf: 510 RouteHit: No
--------------------    Host Route    ----------------------------
vr: 0 2009::1/128 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0 2012::1/128 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
--------------------    IPv6 65-128 pfxlen    --------------------
vr: 0 2012::/96 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No,
(pkt_count=0, devId/pcl=0/75, 1/88)
vr: 0 2009::/112 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No,
(pkt_count=0, devId/pcl=0/76, 1/89)
```

## History

| Release | Command History |
|---|---|
| 08.0.61 | This command was introduced for the ICX 7150 and output was modified. |

# show hardware mac-entry

Displays the hardware information for a specified MAC address or device.

## Syntax

**show hardware mac-entry** [ **device** *device_id* | *mac_address* ]

## Parameters

**device *device_id*** Specifies the hardware device number.
***mac_address*** Specifies a MAC address to be displayed.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example shows command output for hardware device 0. The MAC address and VLAN ID of the device are displayed.

```
ICX7750-26Q# show hardware mac-entry device 0
Total number of entries will be printed at the end of the prints
mac=00e0.5200.0000 vlan=4094 modid=0 port=0 Static COS(src=7,dst=7) CPU Group=(BCM_L2_XXX: 0x4020)
Total number of FDB entries displayed:1
```

## History

| Release | Command History |
|---|---|
| FastIron release 08.0.00a | This command was introduced. |

# show hardware nexthop usage

Displays a summary of the Layer 3 unicast hardware next-hop usage.

## Syntax

**show hardware nexthop usage**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

The **show hardware nexthop usage** command displays the total, available, and used unicast next-hop entries, programmed in hardware for a specific unit in a stack. The next-hop usage summary includes unicast IPv4 and unicast IPv6 entries.

## Examples

The following example displays the hardware next-hop usage entries for the unit (unit 2).

```
device# show hardware nexthop usage
-------------------------------------------------------------------------------
Stack unit: 2
Unicast Nexthop (IPv4+IPv6) entries:
-------------------------------------------------------------------------------
Total           =    13312
Available       =    11264
Used            =     2048
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show hardware route

Displays the hardware information for Layer IPV4 hardware routes.

## Syntax

**show hardware route** { *ip-address* | *ip-address prefix* | **device** *device-id* | **vrf** *name*}

## Parameters

*ip-address*
Specifies an IPv4 address.

*ip-address prefix*
Specifies an IPv4 network number.

**device** *device_id*
Specifies the hardware device number.

**vrf** *vrf-name*
Specifies a VRF instance.

## Modes

User EXEC mode

## Usage Guidelines

**NOTE**
Can cause high CPU/protocol flap and system instability. Please use show hardware route *ip address* to search a specific IP address.

# Examples

The following example displays sample output for the **show hardware route** command.

```
device> show hardware route device 0

Total number of hardware routes: 20   Device-id:0
Ports in this devices are  25 to 48
vr: 0         30.1.1.2/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       18.18.18.1/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0         20.1.1.1/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       70.70.70.1/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0          8.8.8.1/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       80.80.80.1/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0    170.170.170.2/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0    180.180.180.2/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       23.23.23.2/32 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       10.37.82.0/25 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0         30.1.1.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       18.18.18.0/24 NH: 2051 NH hw: 102051 cmd: FWD, Outgoing vlan: 4092, port: INVALID,mac:
0200.8801.002a, L3 intf: 5420 RouteHit: No
vr: 0         20.1.1.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       70.70.70.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0          8.8.8.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       80.80.80.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0    170.170.170.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0    180.180.180.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0       23.23.23.0/24 NH: 2048 NH hw: 102048 cmd: TRAP, Outgoing vlan: 0, L3 intf: 8191 RouteHit: No
vr: 0          0.0.0.0/0 NH: 2049 NH hw: 102049 cmd: DROP, Outgoing vlan: 4091, L3 intf: 4091 RouteHit:
No
```

The following example displays sample output for the **show hardware route** command for ICX 7150 devices.

```
device> show hardware route device 0

Total number of prefix routes: 8   host_routes:4   Device-id:
0                                          <<< host_routes keyword is added.
Ports in this devices are  25 to 48
vr: 0       10.37.82.0/25 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0       23.23.23.0/24 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0          9.9.9.0/24 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0    150.150.150.0/24 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0    160.160.160.0/24 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0        100.1.1.0/24 NH: 1026 NH hw: 101026 cmd: DROP, Outgoing vlan: 4092, L3 intf: 128 RouteHit:
No
vr: 0         30.1.1.0/24 NH: 1027 NH hw: 101027 cmd: FWD, Outgoing vlan: 4092, port: 1/1/48,mac: cc4e.
24f7.2440, L3 intf: 131 RouteHit: No
vr: 0          0.0.0.0/0 NH: 1025 NH hw: 101025 cmd: DROP, Outgoing vlan: 4091, L3 intf: 510 RouteHit:
No
vr: 0    150.150.150.1/32 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0          9.9.9.1/32 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0       23.23.23.1/32 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
vr: 0    160.160.160.1/32 NH: 1024 NH hw: 101024 cmd: TRAP, Outgoing vlan: 0, L3 intf: 511 RouteHit: No
```

# History

| Release | Command History |
|---|---|
| 08.0.61 | This command was introduced for the ICX 7150 and output was modified. |

# show ikev2

Displays global Internet Key Exchange version 2 (IKEv2) configuration information.

## Syntax

**show ikev2**

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

## Command Output

The **show ikev2** command displays the following information:

| Output field | Description |
|---|---|
| Retry Count | The maximum number of attempts that are permitted to retransmit a message. The range is from 1 through 25. The default value is 5. |
| Max Exchange Time | The maximum setup time (in seconds) for an exchange. The range is from 0 through 300. The default value is 30 seconds. |
| Retransmit Interval | The length of time (in seconds) that an IKEv2 task waits before attempting to resend a packet. The range is from 1 through 60. The default value is 5 seconds. The interval between each resend attempt is increased by the value of the retransmit interval; that is, the retransmit interval increases exponentially. |
| Max SA | The maximum number of IKEv2 SAs that may be on a node. The range is from 1 through 256. The default value is 256. |
| Max SA In Nego | The maximum number of IKEv2 security associations (SAs) that may be "in negotiation" on a node. The range is from 1 through 256. The default value is 256. |
| Total IPSEC Intf | The total number of IPsec tunnel interfaces. |
| Total Peers | The total number of peers. |
| Total IPSEC SA | The total number of IPsec SAs (for the total number of IKEv2 SAs). |
| Total IKE SA | The total number of IKEv2 SAs including SAs in active, constructing, and dying states. |
| Cookie Challenge Number | The threshold for issuing an IKEv2 cookie challenge. A challenge is issued when the number of half-open IKEv2 security associations (SAs) crosses the threshold value. The range is from 1 through 512. Cookie challenge is disabled by default. |
| Http Cert Enable | When HTTP Cert is enabled then HTTP_CERT_LOOKUP_SUPPORTED is sent with the CERT_REQ payload. HTTP Cert is disabled by default. |

# Examples

The following example displays global IKEv2 configuration information.

```
device# show ikev2
IKEv2 Global data:
Retry Count          : 5             Max Exchange Time       : 30
Retransmit Interval  : 5             Max SA                  : 256
Max SA In Nego       : 32            Total IPSEC Intf        : 0
Total Peers          : 0             Total IPSEC SA          : 0
Total IKE SA         : 0             Cookie Challenge Number : 0
NAT-T Support enabled: True          NAT Keepalive           : 5
Http Cert Enable        : False (True/False)
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ikev2 auth-proposal

Displays configuration information about Internet Key Exchange version 2 (IKEv2) authentication proposals.

## Syntax

**show ikev2 auth-proposal** [ *name* ]

## Parameters

*name*

Specifies the name of an IKEv2 authentication proposal.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When an IKEv2 authentication proposal is not specified, this command displays information about all configured authentication proposals.

## Command Output

The **show ikev2 auth-proposal** command displays the following information:

| Output field | Description |
|---|---|
| Ikev2 Auth-Proposal | The name of an IKEv2 authentication proposal. |
| Local Auth Method | Local authentication method. |
| Remote Auth Method | Remote authentication method. |
| pre-share-key | Pre-shared key (the encrypted format is displayed). |

## Examples

The following example displays information about the IKEv2 authentication proposal configuration.

```
device# show ikev2 auth-proposal
================================================================================
Ikev2 Auth-Proposal : def-ike-auth-prop
Local Auth Method   : pre_shared
Remote Auth Method  : pre_shared
pre-share-key       : $QG5HTT1Ebk1TVW5NLWIhVW5ATVMhLS0rc1VA
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ikev2 policy

Displays configuration information about Internet Key Exchange version 2 (IKEv2) policies.

## Syntax

**show ikev2 policy** [ *policy-name* ]

## Parameters

*policy-name*

Specifies the name of an IKEv2 policy.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When a policy is not specified, this command displays information about all IKEv2 policies.

## Command Output

The **show ikev2 policy** command displays the following information:

| Output field | Description |
|---|---|
| Name | The name of an IKEv2 policy. |
| vrf | The front-door VRF (fvrf) to match for the policy. |
| Local address/Mask | The local IP address to match for the policy. |
| Proposal | The IKEv2 proposal that is configured for the policy. |
| Ref Count | The number of IPsec profiles that refer to this IKEv2 policy. |

## Examples

The following example displays information about all configured IKEv2 policies.

```
device# show ikev2 policy

   Name                : ike_policy_red
   vrf                 : any
   Local address/Mask  : 0.0.0.0/0.0.0.0
   Proposal            : ike_proposal_red
   Ref Count           : 0

   Name                : def-ike-policy
   vrf                 : any
   Proposal            : def-ike-prop
   Ref Count           : 0
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ikev2 profile

Displays configuration information about Internet Key Exchange version 2 (IKEv2) profiles.

## Syntax

**show ikev2 profile** [ *profile-name* ]

## Parameters

*profile-name*

Specifies the name of an IKEv2 profile.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When a profile is not specified, this command displays information about all IKEv2 profiles.

## Command Output

The **show ikev2 profile** command displays the following information:

| Output field | | Description |
|---|---|---|
| IKEv2 Profile | | The IKEv2 profile name. |
| Auth Profile | | The authentication profile for this IKEv2 profile. |
| Match Criteria | | |
| Inside VRF | | The VRF name. |
| | Local | The local system ID that is compared with the received payload during negotiation. Permitted ID formats are:<br>• address—An IPv4 address<br>• fqdn—A fully qualified domain name, for example, router1.example.com<br>• email—An email address, for example, test@test.com<br>• key-id—A key ID |
| | Remote | Remote system ID that is compared with the received payload during negotiation. Permitted ID formats are:<br>• address—An IPv4 address<br>• fqdn—A fully qualified domain name, for example, router1.example.com<br>• email—An email address, for example, test@test.com<br>• key-id—A key ID |
| Local Identifier | | Local system ID that is sent with the payload during negotiation. Permitted ID formats are:<br>• address—An IPv4 address. |

| Output field | Description |
|---|---|
| | • fqdn—A fully qualified domain name, for example, router1.example.com.<br>• email—An email address, for example, test@test.com.<br>• key-id—A key ID. |
| Remote Identifier | Remote system ID. Permitted ID formats are:<br>• address—An IPv4 address.<br>• fqdn—A fully qualified domain name, for example, router1.example.com.<br>• email—An email address, for example, test@test.com.<br>• key-id—A key ID. |
| Lifetime | The IKEv2 SA lifetime (in minutes). This is also known as the rekey time. |
| Keepalive Check | The interval, in seconds, between the IKEv2 messages sent to detect a dead peer. |
| Initial contact | The initial contact configuration status. When a device reboots, peer devices may have security associations (SAs) that are no longer valid. When initial contact is enabled, an initial contact message is sent to ensure that old security associations (SAs) on the peer are deleted. |
| Ref Count | Number of IPsec profiles that refer to this IKEv2 profile. |

# Examples

The following example displays configuration information for an IKEv2 profile named prof_mktg.

```
device# show ikev2 profile ipsec_tunnel_1

IKEv2 Profile       : ipsec_tunnel_1
Auth Profile        : ipsec_tunnel_1
Match Criteria      :
Inside VRF         : vrf1
  Local:
   email ipsec_tunnel_1@example.com
  Remote:
   email ipsec_tunnel_1@example.com
Local Identifier    : email ipsec_tunnel_1@example.com
Remote Identifier   : email ipsec_tunnel_1@example.com
Lifetime            : 2592000 sec
Keepalive Check     : 10 sec
Initial contact     : yes
Ref Count           : 1
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ikev2 proposal

Displays configuration information about Internet Key Exchange version 2 (IKEv2) proposals.

## Syntax

**show ikev2 proposal** [ *name* ]

## Parameters

*name*

Specifies the name of an IKEv2 proposal.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When an IKEv2 proposal is not specified, this command displays configuration information for all IKEv2 proposals.

## Command Output

The **show ikev2 proposal** command displays the following information:

| Output field | Description |
|---|---|
| Name | The name of an IKEv2 proposal. |
| Encryption | The encryption algorithms that are configured for the proposal. |
| Integrity | The integrity algorithms that are configured for the proposal. |
| PRF | The pseudorandom function algorithms that are configured for the proposal. |
| DH Group | The Diffie-Hellman groups that are configured for the proposal. |
| Ref Count | The number of IPsec profiles that refer to this IKEv2 proposal |

## Examples

The following example shows how to display information about the IKEv2 proposal configuration.

```
device# show ikev2 proposal

     Name       : def-ike-prop
     Encryption : aes256
     Integrity  : sha384
     PRF        : sha384
     DH Group   : 384_ECP/Group 20
     Ref Count  : 2
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ikev2 sa

Displays configuration information about current Internet Key Exchange version 2 (IKEv2) security associations (SAs).

## Syntax

**show ikev2 sa** [ **detail** ]

**show ikev2 sa fvrf** *vrf-name*

**show ikev2 sa interface** *tunnel-port* [ **detail** ]

**show ikev2 sa ipv4**

**show ikev2 sa ipv6**

**show ikev2 sa local** { *ip-address* | *ipv6-address* } [ **detail** ]

**show ikev2 sa remote** { *ip-address* | *ipv6-address* } [ **detail** ]

## Parameters

**detail**
Specifies the display of detailed information.

**fvrf** *vrf-name*
Specifies the name of a forwarding VRF.

**interface** *tunnel-port*
Specifies a tunnel port number.

**ipv4**
Specifies IPv4 connections.

**ipv6**
Specifies IPv6 connections.

**local** *ip-address*
Specifies a local interface.

*ip-address*
Specifies an IPv4 address.

*ipv6-address*
Specifies an IPv6 address.

**remote**
Specifies a remote interface.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When the **detail** option is omitted, only the basic SA information is displayed.

# Command Output

The **show ikev2 sa** command displays the following information:

| Output field | Description |
|---|---|
| Total SA | The total number of IKEv2 SAs, that is; SAs that are in active, constructing, and dying states. |
| Active SA | The number of IKEv2 SAs in an active state. |
| Constructing SA | The number of IKEv2 SAs in a constructing state. |
| Dying SA | The number of IKEv2 SAs in an dying state. |
| tnl-id | The tunnel interface ID for the IKEv2 SA. |
| local | The local address of the tunnel. |
| remote | The remote address of the tunnel. |
| status | The IKEv2 SA state. |
| vrf(i) | The base or internal VRF for the IKEv2 tunnel. |
| vrf(f) | The front-end (customer end) VRF for the IKEv2 tunnel. |
| Role | The role of the device (initiator, responder). |
| Local SPI | The local security parameter index (SPI) for the IKEv2 SA. |
| Remote SPI | The remote SPI for the IKEv2 SA. |
| Profile | The IKEv2 profile for the session. |
| Policy | The IKEv2 policy for the session. |
| Auth Proposal | The IKEv2 authentication proposal for the session. |

# Examples

The following example displays information about the current SA configuration, in which there are four active SAs.

```
device# show ikev2 sa

Total SA : 4
Active SA: 4    : Constructing SA:0     : Dying SA:0
------------------------------------------------------------------------------
tnl-id   local           remote          status   vrf(i)        vrf(f)
------------------------------------------------------------------------------
tnl 18   10.18.3.4/500   10.18.3.5/500   active   default-vrf   default-vrf
tnl 22   10.22.3.4/500   10.22.3.5/500   active   default-vrf   default-vrf
tnl 19   10.19.3.4/500   10.19.3.5/500   active   default-vrf   default-vrf
tnl 20   10.20.3.4/500   10.20.3.5/500   active   default-vrf   default-vrf
```

The following example displays detailed IKEv2 SA information.

```
device# show ikev2 sa detail

Total SA : 1
Active SA: 1    : Constructing SA:0     : Dying SA:0
--------------------------------------------------------------------------------
tnl-id  Local           Remote          Status       Vrf(i)      Vrf(f)
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
tnl 1   10.1.41.1       10.4.41.1       Active       vrf1        vrf2
--------------------------------------------------------------------------------
Role                : Initiator
Local SPI           : 0x6fb19219160c7d71     Remote SPI: 0xde1b24e5764f311e
Profile             : p1
Policy              : ipsec_tunnel_1
Auth Proposal       : p1
```

The following example displays IKEv2 SA information, including information about IPv6 connections.

```
device# show ikev2 sa
Total SA : 7
Active SA: 7    : Constructing SA:0     : Dying SA:0
--------------------------------------------------------------------------------------------------------------
---------------
tnl-id  Local            Remote            Status       Vrf(i)      Vrf(f)
--------------------------------------------------------------------------------------------------------------
---------------
tnl 8   2220::1          5002::2           Active       default-vrf default-vrf
tnl 7   1110::1          5002::2           Active       default-vrf default-vrf
tnl 1   1000::1          1004::2           Active       default-vrf default-vrf
tnl 4   120.1.1.1        110.1.1.1         Active       default-vrf default-vrf
tnl 11  1000::1          1003::2           Active       default-vrf default-vrf
tnl 9   3330::1          5002::2           Active       default-vrf default-vrf
tnl 3   100.1.1.1        104.1.1.2         Active       default-vrf default-vrf
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support was added for IPv6. |

# show ikev2 session

Displays Internet Key Exchange version 2 (IKEv2) session information that includes rekeys and other negotiated information.

## Syntax

**show ikev2 session** [ *local-spi-id* | **detail** ]

## Parameters

*local-spi-id*

Specifies the security parameter index (SPI) for the IKEv2 session.

**detail**

Specifies the display of detailed information about IKEv2 sessions.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

## Command Output

The **show ikev2 session** command displays the following information:

| Output field | Description |
|---|---|
| IKE count | The total number of IKEv2 security associations (SAs). |
| Child Sa Count | The total number of IPsec security associations (SAs). |
| tnl-id | The tunnel interface ID for the IKEv2 SA. |
| local | The local address of the tunnel. |
| remote | The remote address of the tunnel. |
| status | The IKEv2 SA state. |
| vrf(i) | The base or internal VRF for the IKEv2 tunnel. |
| vrf(f) | The front-end (customer end) VRF for the IKEv2 tunnel. |
| Encr | The encryption algorithm used by this session after IKEv2 negotiations. |
| Hash | The hashing algorithm used by this session after IKEv2 negotiations. |
| DH Grp | The Diffie-Hellman (DH) group used by this session after IKEv2 negotiations. |
| Auth | The authentication method used by this session after IKEv2 negotiations. |
| PRF | The pseudorandom function (PRF) used by this session after IKEv2 negotiations. |
| Local spi | The local security parameter index (SPI) for the session. |
| Remote spi | The remote SPI for the session. |

| Output field | Description |
|---|---|
| Life/Active Time | The configured IKEv2 rekey time and the time left until the next rekey. |
| Rekey count Local | The total number of session key changes for the IKEv2 SA that were initiated by the local device. |
| Rekey count Remote | The total number of session key changes for the IKEv2 SA that were initiated by the remote device. |
| Status Description | The IKEv2 SA state. |
| Initiator id | The initiator identity for the IKEv2 SA. |
| Responder id | The responder identity for the IKEv2 SA. |
| no Exchange in progress | Indicates that this session is not in an exchange state. |
| next request message id | The next message ID for the session. |
| Keepalive timer | The interval between IKEv2 messages that are sent to detect if a peer is still alive. |
| Total keepalive sent | The total number of "keepalive" messages sent for the session. |
| Total keepalive received | The total number of "keepalive" messages received for the session. |
| Total Bytes sent | The total number of bytes sent in the session. |
| Total Bytes Received | The total number of bytes received in the session. |
| Time past since last msg | The elapsed time since the last message. |
| NAT-T | Network Address Translation (NAT) configuration status. |
| Child Sa | IPsec SA details. |
| id | The numeric identifier for an IPsec SA. |
| Local selector | The local traffic selector. |
| Remote selector | The remote traffic selector. |
| ESP SPI IN/OUT | The IPsec SPI for ingress and the SPI for egress. |
| Encryption | The encryption algorithm used by the session. |
| ICV Size | The size of the integrity check value (ICV) for the encryption algorithm. |
| Esp_hmac | The hashed message authentication code (HMAC) algorithm used by the session. |
| Authentication | The authentication algorithm used by the session. |
| DH Group | The Diffie-Hellman (DH) group used by the authentication algorithm. |
| Mode | The Encapsulating Security Protocol (ESP) mode for the session. |
| Rekey count Local | The total number of changes to the IPsec SA session key initiated by the local device. |
| Rekey count Remote | The total number of changes to the IPsec SA session key initiated by the remote device. |

# Examples

The following example displays IKEv2 session information.

```
device# show ikev2 session

IKE count:1, Child Sa Count:2
tnl-id     local        remote       status     vrf(i)        vrf(f)
-------------------------------------------------------------------------------
tnl 18     10.18.3.4    10.18.3.5    active     default-vrf   default-vrf
-------------------------------------------------------------------------------
    Encr: aes-cbc-256, Hash: sha384, DH Grp:384_ECP/Group 20, Auth: pre_shared
    PRF: sha384
    Is Initiator: Yes
    Local spi   : 0xe115847e85ad667b       Remote spi: 0x7bb5ee3b6074a4b4
    Life/Active Time: 2592000/534 sec
    Rekey count Local: 0       Rekey count Remote: 2
Child Sa:
 id 1
       Local selector  0.0.0.0/0 - 255.255.255.255
       Remote selector 0.0.0.0/0 - 255.255.255.255
       ESP SPI IN/OUT: 0xb278/0x7935
       Encryption: aes-gcm-256, ICV Size: 16 octects, Esp_hmac: Null
       Authetication: null  DH Group:none , Mode: tunnel
       Rekey count Local: 0       Rekey count Remote: 2
```

The following example displays detailed IKEv2 session information.

```
device# show ikev2 session detail

IKE count:4, Child Sa Count:8
tnl-id     local        remote       status     vrf(i)        vrf(f)
-------------------------------------------------------------------------------
tnl 18     10.18.3.4    10.18.3.5    active     default-vrf   default-vrf
-------------------------------------------------------------------------------
    Encr: aes-cbc-256, Hash: sha384, DH Grp:384_ECP/Group 20, Auth: pre_shared
    PRF: sha384
    Local spi   : 0xe115847e85ad667b       Remote spi: 0x7bb5ee3b6074a4b4
    Life/Active Time: 2592000/614 sec
    Rekey count Local: 0       Rekey count Remote: 2
    Status Description: active
    Initiator id: address 18.3.3.4    Responder id: address 18.3.3.5
    no Exchange in progress
    next request message id=4
    Keepalive timer: 300 seconds, retry 0
        Total keepalive sent: 2
        Total keepalive received: 0
        Total Bytes sent    : 524   Total Bytes Received   : 672
    Time past since last msg: 14
    NAT-T is not detected
Child Sa:
 id 1
       Local selector  0.0.0.0/0 - 255.255.255.255
       Remote selector 0.0.0.0/0 - 255.255.255.255
       ESP SPI IN/OUT: 0xb278/0x7935
       Encryption: aes-gcm-256, ICV Size: 16 octects, Esp_hmac: Null
       Authetication: null  DH Group:none , Mode: tunnel
       Rekey count Local: 0       Rekey count Remote: 2
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ikev2 statistics

Displays statistical information about Internet Key Exchange version 2 (IKEv2).

## Syntax

**show ikev2 statistics**

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

## Command Output

The **show ikev2 statistics** command displays the following information:

| Output field | | Description |
|---|---|---|
| Total IKEv2 SA Count active | | The total number of IKEv2 security associations (SAs) in an active state. |
| Incoming IKEv2 Requests | | The number of IKEv2 SAs (accepted and rejected) initiated by the peer device. |
| Outgoing IKEv2 Requests | | The number of IKEv2 SAs initiated by the local device. |
| | Accepted | The total number of outgoing IKEv2 SAs that were accepted. |
| | Rejected | The total number of outgoing IKEv2 SAs that were rejected. |
| | Rejected due to no cookie | The total number of outgoing IKEv2 SAs that were rejected due to no cookie. |
| IKEv2 Packet Statistics | | |
| | Total Packets Received | The total number of packets received. |
| | Total Packets Transmitted | The total number of packets transmitted. |
| | Total Packets Retransmitted | The total number of packets retransmitted. |
| | Total Failed Transmission | The total number of packets where transmission failed. |
| | Total Pending Packets | The total number of packets to be transmitted. |
| | Total Buffer Failed | The total number of packets where transmission failed due to a buffer issue. |
| | Total Keepalive Received | The total number of IKEv2 keepalive messages received. |
| | Total Keepalive Transmitted | The total number of IKEv2 keepalive messages transmitted. |
| IKEv2 Error Statistics | | |
| | Unsupported Payload | The total number of IKEv2 packets received with an unsupported payload. |
| | Invalid IKE SPI | The total number of IKEv2 packets received with an invalid security parameter index (SPI). |
| | Invalid Version | The total number of IKEv2 packets received with an invalid version. |
| | Invalid Syntax | The total number of IKEv2 packets received with invalid syntax. |
| | Negotiation Timeout | The total number of IKEv2 sessions deleted due to dead peer detection (DPD) or negotiation timeouts. |
| | No Policy | The total number of IKEv2 sessions deleted or rejected due to a policy issue. |

| Output field | | Description |
|---|---|---|
| | No Protection Suite | The total number of IKEv2 sessions deleted or rejected due to a protection suite issue. |
| | Policy Error | The total number of IKEv2 sessions deleted or rejected due to policy error. |
| | IKE Packet Error | The total number of IKEv2 or IPsec packets received with a packet error. |
| | Discard Policy | The total number of IKEv2 or IPsec sessions deleted or rejected due to a policy error or mismatch. |
| | Proposal Mismatch | The total number of IKEv2 or IPsec packets sent or received with a proposal mismatch. |
| | Invalid Selectors | The total number of IKEv2 or IPsec packets sent or received with invalid selectors. |
| | Internal Error | The total number of IKEv2 or IPsec packets sent or received with an internal error. |
| | SA Overflow | The total number of times the maximum SA count was reached. |
| | IKE SA Overflow | The number of times the maximum IKEv2 SA count was reached. |
| | IPSEC SA Overflow | The number of times the maximum IPsec SA count was reached. |
| | Authentication Failed | The total number of IKEv2 or IPsec packets sent or received when authentication failed. |
| | Others | The total number of IKEv2 or IPsec packets sent or received with other error types. |
| | Number of HW-SPI Add write | The number of times the creation of an IPsec SPI was written to the hardware. |
| | Number of HW-SPI Delete | The number of times the deletion of an IPsec SPI was written to the hardware. |

# Examples

The following example displays IKEv2 statistics.

```
device# show ikev2 statistics

Total IKEv2 SA Count active: 0
Incoming IKEv2 Requests: Accepted: 0 Rejected: 0
Outgoing IKEv2 Requests: 0
    Accepted: 0 Rejected: 0 Rejected due to no cookie: 0
IKEv2  Packet Statistics:
    Total Packets Received     : 0
    Total Packets Transmitted  : 2
    Total Packets Retransmitted: 0
    Total Failed Transmission  : 0
    Total Pending Packets      : 0
    Total Buffer Failed        : 0
    Total Keepalive Received   : 0
    Total Keepalive Transmitted: 0
IKEv2 Error Statistics:
    Unsupported Payload   : 0        Invalid IKE SPI  : 0
    Invalid Version       : 0        Invalid Syntax   : 0
    Negotiation Timeout   : 0        No Policy        : 0
    No Protection Suite   : 0        Policy Error     : 0
    IKE Packet Error      : 1        Discard Policy   : 0
    Proposal Mismatch     : 0        Invalid Selectors: 0
    Internal Error        : 0        SA Overflow      : 0
    IKE SA Overflow       : 0        IPSEC SA Overflow: 0
    Authentication Failed : 0        Others           : 0
    Number of HW-SPI Add write : 0   Number of HW-SPI Delete write: 0
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show inline power

Displays the inline power capacity, power allocation, power consumption, and power priority details for Power over Ethernet (PoE) ports.

## Syntax

**show inline power** [ **stack-unit** | **stack/slot/port** [ **debug-info** ] ]

## Parameters

**stack-unit**
Displays inline power information for the specified stack unit.

**stack/slot/port**
Displays inline power information for a specific interface.

**debug-info**
Displays inline power debugging information for the specified interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

Use this command to view details about PoE power usage.

You can view the PoE operational status for the entire device, for a specific PoE module, or for a specific interface.

## Command Output

The **show inline power** command displays the following information.

| Output field | Description |
|---|---|
| Power Capacity | The total PoE power supply capacity and the amount of available power (current free) for PoE-power-consuming devices. Both values are shown in milliwatts. |
| Power Allocations | The number of times the device fulfilled PoE requests for power. |
| Port | The slot number and port number. |
| Admin State | Specifies whether Power over Ethernet has been enabled on the port:<br>• On: The **inline power** command was issued on the port.<br>• Off: The **inline power** command has not been issued on the port. |
| Oper State | Shows the status of inline power on the port:<br>• On: The PoE power supply is delivering inline power to the powered device (PD). |

| Output field | Description |
|---|---|
| | • Off: The PoE power supply is not delivering inline power to the PD.<br>• Non-PD: Identifies the ports connected to nonpowered devices.<br>• Denied: The port is in standby mode (waiting for power) because the device does not currently have enough available power for the port.<br><br>**NOTE**<br>When you enable a port using the CLI, it may take 12 or more seconds before the operational state of that port is displayed correctly in the **show inline power** output. |
| Power Consumed | The number of current, actual milliwatts that the PD is consuming. |
| Power Allocated | The number of milliwatts allocated to the port. This value is either the default or configured maximum power level or the power class that was automatically detected by the device. |
| PD Type | The type of PD connected to the port:<br>• 802.3at: The PD connected to this port is 802.3at-compliant.<br>• 802.3af: The PD connected to this port is 802.3af-compliant.<br>• Legacy: The PD connected to this port is a legacy product (not 802.3af-compliant).<br>• N/A: Power over Ethernet is configured on this port, and one of the following is true:<br>– The device connected to this port is a nonpowered device.<br>– No device is connected to this port.<br>– The port is in standby or denied mode (waiting for power).<br><br>**NOTE**<br>Although not 802.3af-compliant, some legacy products may show the PD type as 802.3af. |
| PD Class | Determines the maximum amount of power that a PD receives. This field can also be "n/a" meaning that the device attached to the port cannot advertise its power class.<br><br>**NOTE**<br>If an 802.3at PD with a class 4 value is connected to a Ruckus ICX switch, the switch must be running FastIron release 08.0.20 or later to be able to perform the necessary power negotiations. |
| Pri | The port inline power priority, which determines the order in which the port receives power while in standby mode (waiting for power). Ports with a higher priority receive power before ports with a lower priority. This value can be one of the following:<br>• 3: Low priority<br>• 2: High priority<br>• 1: Critical priority |
| Fault/Error | If applicable, the fault or error that occurred on the port:<br>• critical temperature — The PoE chip temperature limit rose above the safe operating level, thereby powering down the port.<br>• detection failed: discharged capacitor — The port failed capacitor detection (legacy PD detection) because of a discharged capacitor. This can occur when connecting a non-PD on the port.<br>• detection failed: out of range capacitor — The port failed capacitor detection (legacy PD detection) because of an out-of-range capacitor value. This can occur when connecting a non-PD on the port.<br>• internal h/w fault — A hardware problem hindered port operation.<br>• lack of power: The port shut down due to lack of power. |

| Output field | Description |
|---|---|
| | • main supply voltage high — The voltage was higher than the maximum voltage limit, thereby tripping the port.<br>• main supply voltage low — The voltage was lower than the minimum voltage limit, thereby tripping the port.<br>• overload state — The PD consumed more power than the maximum limit configured on the port, based on the default configuration, user configuration, or CDP configuration.<br>• over temperature — The port temperature rose above the temperature limit, thereby powering down the port.<br>• PD DC fault — A succession of underload and overload states, or a PD DC/DC fault, caused the port to shut down.<br>• short circuit — A short circuit was detected on the port delivering power.<br>• underload state — The PD consumed less power than the minimum limit specified in the 802.3af standard.<br>• voltage applied from ext src — The port failed capacitor detection (legacy PD detection) because the voltage applied to the port was from an external source. |
| Total | The total power in milliwatts being consumed by all PDs connected to the interface module and the total power in milliwatts allocated to all PDs connected to the interface module. |
| Grand Total | The total number of current, actual milliwatts being consumed by all PDs connected to the PoE device and the total number of milliwatts allocated to all PDs connected to the PoE device. |

# Examples

The following is sample output from the **show inline power** command.

```
device# show inline power

Power Capacity:        Total is 2160000 mWatts. Current Free is 18800 mWatts.
Power Allocations:     Requests Honored 769 times

... some lines omitted for brevity ...

Port   Admin Oper   --Power(mWatts)--  PD Type   PD Class Pri Fault/Error
       State State Consumed Allocated
-------------------------------------------------------------------------
1/1/1  On    On     5070     9500      802.3af   n/a      3   n/a
1/1/2  On    On     1784     9500      Legacy    n/a      3   n/a
1/1/3  On    On     2347     9500      802.3af   n/a      3   n/a
1/1/4  On    On     2441     9500      Legacy    n/a      3   n/a
1/1/5  On    On     6667     9500      802.3af   Class 3  3   n/a
1/1/6  On    On     2723     9500      802.3af   Class 2  3   n/a
1/1/7  On    On     2347     9500      802.3af   n/a      3   n/a
1/1/8  On    On     2347     9500      802.3af   n/a      3   n/a
1/1/9  On    On     2347     9500      802.3af   n/a      3   n/a
1/1/10 On    On     4976     9500      802.3af   Class 3  3   n/a
1/1/11 On    On     4882     9500      802.3af   Class 3  3   n/a
1/1/12 On    On     4413     9500      802.3af   Class 1  3   n/a
1/1/13 On    On     7793     9500      802.3af   n/a      3   n/a
1/1/14 On    On     7512     9500      802.3af   n/a      3   n/a
1/1/15 On    On     8075     9500      802.3af   n/a      3   n/a
1/1/16 On    On     4131     9500      802.3af   Class 1  3   n/a
1/1/17 On    Non-PD 0        0         n/a       n/a      3   n/a
1/1/18 On    Non-PD 0        0         n/a       n/a      3   n/a
1/1/19 On    Off    0        30000     n/a       n/a      3   n/a
1/1/20 On    Off    0        30000     n/a       n/a      3   n/a
1/1/21 On    Non-PD 0        0         n/a       n/a      3   n/a
1/1/22 On    Non-PD 0        0         n/a       n/a      3   n/a
1/1/23 On    Non-PD 0        0         n/a       n/a      3   n/a
1/1/24 On    Non-PD 0        0         n/a       n/a      3   n/a
-------------------------------------------------------------------------
Total                     137367         242000

... some lines omitted for brevity...

Grand Total             1846673        2127400
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |
| 8.0.50 | A new operating state was added (Non-PD). |

# show inline power debug-info

Displays inline power debug information.

## Syntax

**show inline power debug-info** [ stack_unit | unit/slot/port ]

## Parameters

stack_unit
: Displays inline power debug information for the specified stack unit or SPX unit ID.

unit/slot/port
: Displays inline power debug information for the specified interface.

## Modes

Privileged exec mode

## Usage Guidelines

The command prints the complete output of **show inline power** port-id plus the last five hardware port states, and the last five software port states.

## Examples

Use the following command to display inline power information that is of use in debugging the configuration.

```
device# (config-if-e1000-8/1/3)# show inline power debug-info 8/1/3

 Port   Admin   Oper    ---Power(mWatts)---  PD Type  PD Class  Pri  Fault/
        State   State   Consumed  Allocated                        Error
-------------------------------------------------------------------------
  8/1/3 On      On          8100      38115  802.3at  Class 4     3  n/a
 hwEvLatch:0, afAtPoh:2, pair4En:1
 Last 5 HW port status:
        1:0x02 PD Detected on 4-Pair lines             2:0x1A User OFF
        3:0x02 PD Detected on 4-Pair lines             4:0x1A User OFF
        5:0x02 PD Detected on 4-Pair lines

Max Power Capability for 2pair PD :45000 mWatts
Highest Power Requested by PD Through LLDP/CDP :38114 mWatts
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.61 | The output was enhanced with POE overdrive information. |

# show inline power detail

Displays detailed information about the PoE power supplies installed in a PoE device.

## Syntax

**show inline power detail** [ *stack-unit* | **debug-info** *stack-unit* ]

## Parameters

*stack-unit*
Displays detailed inline power information for the specified stack unit.

debug-info
Displays detailed debug information.

## Modes

User EXEC mode

## Usage Guidelines

You can view the PoE operational status for the entire device, for a specific PoE module, or for a specific interface.

## Command Output

The **show inline power detail** command displays the following information.

| Output field | Description |
|---|---|
| Max Curr: | The number of milliwatts available for the unit in the stack, not the entire stack. This value is either the default or configured maximum power level, or the power class that was automatically detected by the device. |
| Voltage | The number of Volts allocated to the stack. |
| Capacity | The total PoE power supply capacity and the amount of available power (current free) for PoE-power-consuming devices. Both values are shown in milliwatts. |
| Firmware Version | Returns firmware version information. |
| Hardware Version | Returns hardware version information. |
| Power Allocations | The number of times the device fulfilled PoE requests for power. |
| Cumulative Port State Data | Shows the number of ports with a particular status/configuration. |
| Cumulative Port Power Data | Shows the port power consumption and allocation. |
| Device Status | The status of the device:<br>• Failed: No (bad or unreachable) PoE device found (bootup time)<br>• Good: OK - Expected PoE device Found (Zone1, no VOP)<br>• Revived: Device is currently refreshed<br>• n/a: Reserved<br>• Lost: Device lost or different from expected (while in operation)<br>• VOP-Sev1: Device error 1 (Zone2, VOP severity 1) |

| Output field | Description |
|---|---|
| | • VOP-Sev2: Device error 1 (Zone2, VOP severity 2)<br>• VOP-Sev3: Device error 1 (Zone2, VOP severity 3)<br>• VmErr: Device Vmain Error<br>• Vm2vErr: Device Vmain < System AVG Vmain by 2v or more<br>• Recovered: OK - Expected PoE device Found (Zone3, recovered from VOP) |

# Examples

The following is an example of **show inline power detail** command output for an ICX 7150 device.

```
device# show inline power detail
Power Supply Data On unit 1:
++++++++++++++++++
Power Supply Data:
++++++++++++++++++

power supply 1 is not present
Power Supply #2:
        Max Curr:       13.8 Amps
        Voltage:        54.0 Volts
        Capacity:       748 Watts


POE Details Info. On Unit 1 :


General PoE Data:
+++++++++++++++++

Firmware
Version
----------------
01.6.7 Build 013

Hardware
Version
-----------------
V1R3

Cumulative Port State Data:
+++++++++++++++++++++++++++

#Ports    #Ports     #Ports   #Ports    #Ports       #Ports     #Ports
Admin-On  Admin-Off  Oper-On  Oper-Off  Off-Denied   Off-No-PD  Off-Fault
-------------------------------------------------------------------------
30        2          7        25        0            23         2
Cumulative Port Power Data:
+++++++++++++++++++++++++++

#Ports  #Ports  #Ports        Power       Power
Pri: 1  Pri: 2  Pri: 3  Consumption  Allocation
-----------------------------------------------
1       0       29         43.900 W   470.000 W
```

The following example shows sample output from the **show inline power detail** command, including information about the device status, when the **debug-info** keyword is used.

```
device> show inline power detail debug-info

Power Supply Data On unit 1:
++++++++++++++++++

Power Supply Data:
++++++++++++++++++
power supply 1 is not present
Power Supply #2:
        Max Curr:       13.9 Amps
        Voltage:        54.0 Volts
        Capacity:       748 Watts
        PoePower:       748 Watts

POE Details Info. On Unit 1 :

General PoE Data:
+++++++++++++++++
```

```
Firmware
Version
----------------
02.1.1 Build 002
Hardware
Version
----------------
V2R2
Note: Number of PoE Devices:9. This number of LSBs should be zero
                in devFaultMap, devTempOff,devTempAlarm
First System Status:
               cpuStatus1 (fwDnldReq:0, errController:0), cpuStatus2(errMemory:0),
               factoryDefault:1, genInternalErr:0, privateLabel:0, userByte:ff,
               devFaultMap:0, devTempOff: 0, devTempAlarm: 0, intReg:2f2c
Latest System Status:
               cpuStatus1 (fwDnldReq:0, errController:0), cpuStatus2(errMemory:0),
               factoryDefault:0, genInternalErr:0, privateLabel:0, userByte:0,
               devFaultMap:0, devTempOff: 0, devTempAlarm: 0, intReg:2f2c
Device HW version          : 0:V2R2      1:V2R2      2:V2R2      3:V2R2      4:V2R2      5:V2R2
6:V2R2      7:V2R2      8:V2R2
Device Temperature(deg-C) : 0:40        1:40        2:42        3:44        4:48        5:48
6:46        7:46        8:44
Device Status              : 0:Good      1:Good      2:Good      3:Good      4:VOP-Sev1  5:VOP-Sev1
6:Good      7:Good      8:Good

Cumulative Port State Data:
++++++++++++++++++++++++++
#Ports    #Ports     #Ports   #Ports    #Ports       #Ports     #Ports
Admin-On  Admin-Off  Oper-On  Oper-Off  Off-Denied   Off-No-PD  Off-Fault
-------------------------------------------------------------------------
48        0          10       38        0            38         0

Cumulative Port Power Data:
++++++++++++++++++++++++++
#Ports  #Ports  #Ports        Power       Power
Pri: 1  Pri: 2  Pri: 3  Consumption  Allocation
-----------------------------------------------
0       0       48         42.700 W   507.515 W


Power Supply Data On unit 6:
++++++++++++++++++

Power Supply Data:
++++++++++++++++++
Power Supply #1:
        Max Curr:       13.9 Amps
        Voltage:        54.0 Volts
        Capacity:       748 Watts
        PoePower:       748 Watts
power supply 2 is not present

POE Details Info. On Unit 6 :

General PoE Data:
++++++++++++++++++
Firmware
Version
----------------
02.1.1 Build 002
Hardware
Version
----------------
V2R4
Note: Number of PoE Devices:9. This number of LSBs should be zero
                in devFaultMap, devTempOff,devTempAlarm
First System Status:
               cpuStatus1 (fwDnldReq:0, errController:0), cpuStatus2(errMemory:0),
               factoryDefault:1, genInternalErr:0, privateLabel:0, userByte:ff,
               devFaultMap:0, devTempOff: 0, devTempAlarm: 0, intReg:3304
Latest System Status:
               cpuStatus1 (fwDnldReq:0, errController:0), cpuStatus2(errMemory:0),
```

```
               factoryDefault:0, genInternalErr:0, privateLabel:0, userByte:0,
               devFaultMap:0, devTempOff: 0, devTempAlarm: 0, intReg:3304
Device HW version          : 0:V2R4      1:V2R4      2:V2R4      3:V2R4      4:V2R4      5:V2R4
6:V2R4      7:V2R4      8:V2R4
Device Temperature(deg-C) : 0:42        1:40        2:42        3:44        4:42        5:46
6:40        7:44        8:40
Device Status              : 0:Good      1:Good      2:Good      3:Good      4:Good      5:Good
6:Good      7:Good      8:Good

Cumulative Port State Data:
++++++++++++++++++++++++++
#Ports    #Ports     #Ports    #Ports    #Ports       #Ports     #Ports
Admin-On  Admin-Off  Oper-On   Oper-Off  Off-Denied   Off-No-PD  Off-Fault
-------------------------------------------------------------------------
48        0          11        37        0            36         0

Cumulative Port Power Data:
++++++++++++++++++++++++++
#Ports  #Ports  #Ports        Power       Power
Pri: 1  Pri: 2  Pri: 3  Consumption  Allocation
-----------------------------------------------
0       0       48         46.700 W   445.435 W
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |
| 08.0.50 | Added the **debug-info** option. |
| 08.0.60 | The output field included Hardware Version. |
| 08.0.70d | Information on device status was added to the command output. |
| 08.0.80e | Information on device status was added to the command output. |

# show inline power emesg

Displays a history of Power over Ethernet (PoE) events.

## Syntax

**show inline power emesg** *unit-id count*

## Parameters

*unit-id*

Specifies the number of the unit.

*count*

Number of logged PoE events to print. By default, 2000 PoE events are printed if a count is not specified.

## Modes

Privileged exec mode

## Usage Guidelines

The command prints the last 2000 PoE events from each unit of the system.

## Examples

The following is sample output from the **show inline power emesg** command.

```
device# show inline power emesg 18 16
Log Size: 2000 entries.         Number of entries in use: 2000  for unit 18(full).
Logging is active.
Log printing is requested for last (latest) 16 entries.
+-------+----------------+-----+-----+----------+-----------------------------------------+
|SL Num.|Timestamp       | Sys | Dev | Port     | Event Trace Message                     |
+-------+----------------+-----+-----+----------+-----------------------------------------+
 1      | Jan 23 20:58:00 |  N  | N/A | 18/1/5   | Port is in detection mode (port is off)
 2      | Jan 23 20:58:42 |  N  | N/A | 18/1/5   | Port has a non-standard PD connected and is o
 3      | Jan 23 20:58:46 |  N  | N/A | 18/1/5   | Port is in detection mode (port is off)
 4      | Jan 23 20:59:39 |  N  | N/A | 18/1/13  | Port is off due to overload state
 5      | Jan 23 20:59:51 |  N  | N/A | 18/1/13  | Port is in detection mode (port is off)
 6      | Jan 23 20:59:56 |  N  | N/A | 18/1/13  | Port is off due to overload state
 7      | Jan 23 21:00:07 |  N  | N/A | 18/1/13  | Port is in detection mode (port is off)
 8      | Jan 23 21:00:20 |  N  | N/A | 18/1/13  | Port is off due to overload state
 9      | Jan 23 21:00:30 |  N  | N/A | 18/1/13  | Port is in detection mode (port is off)
 10     | Jan 23 21:01:24 |  N  | N/A | 18/1/13  | Port is off due to overload state
 11     | Jan 23 21:01:32 |  N  | N/A | 18/1/13  | Port is in detection mode (port is off)
 12     | Jan 23 21:02:20 |  N  | N/A | 18/1/5   | Port has a non-standard PD connected and is o
 13     | Jan 23 21:02:23 |  N  | N/A | 18/1/5   | Port is in detection mode (port is off)
 14     | Jan 23 21:02:39 |  N  | N/A | 18/1/13  | Port is off due to overload state
 15     | Jan 23 21:02:46 |  N  | N/A | 18/1/13  | Port is in detection mode (port is off)
 16     | Jan 23 21:03:10 |  N  | N/A | 18/1/13  | Port has a non-standard PD connected and is o
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |
| 08.0.61 | This command was modified. |

# show interfaces ethernet

Displays Ethernet interface information.

## Syntax

**show interfaces ethernet** *stackid**/**slot**/**port*

## Parameters

*stackid / slot / port*

Stack ID number, slot number, and port number for an existing Ethernet interface.

## Modes

Privileged EXEC mode

# Examples

This example shows detailed interface information. Note that the priority flow control (PFC) is shown as enabled and information for the unicast and multicast egress queues is shown separately.

```
device# show interfaces ethernet 1/1/22

10GigabitEthernet1/1/22 is up, line protocol is up
  Port up for 16 minutes 1 seconds
  Hardware is 10GigabitEthernet, address is aabb.ccdd.ef14 (bia aabb.ccdd.ef14)
  Configured speed 10Gbit, actual 10Gbit, configured duplex fdx, actual fdx
  Member of 1 L2 VLANs, port is tagged, port state is FORWARDING
  BPDU guard is Disabled, ROOT protect is Disabled, Designated protect is Disabled
  Link Error Dampening is Disabled
  STP configured to ON, priority is level0, mac-learning is enabled
  ….
  …..
  MTU 1500 bytes
  Priority-Flow-Control is Enabled
  300 second input rate: 37014512 bits/sec, 9036 packets/sec, 0.38% utilization
  300 second output rate: 731174584 bits/sec, 178509 packets/sec, 7.58% utilization
  0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 multicasts, 0 unicasts
  0 input errors, 0 CRC, 0 frame, 0 ignored
  0 runts, 0 giants
  26055807 packets output, 13340529672 bytes, 0 underruns
  Transmitted 0 broadcasts, 98 multicasts, 26055709 unicasts
  0 output errors, 0 collisions
  Relay Agent Information option: Disabled

UC Egress queues:
Queue counters    Queued packets    Dropped Packets
         0                 0             2074860
         1           2349160             2074861
         2           2349163             2074861
         3           2349165             2074860
         4           2349163             2074860
         5           2349165             2074860
         6           5461694              518651
         7           6498353                   0


MC Egress queues:
Queue counters    Queued packets    Dropped Packets
         0                 0                   0
         1                 0                   0
         2                 0                   0
         3                 0                   0
         4                 0                   0
```

This example shows information for an interface that has an ingress profile and an egress profile attached to a port.

```
device(config-if-e40000-1/1/1)# show interfaces ethernet 1/1/1

40GigabitEthernet1/1/1 is up, line protocol is up
  Port up for 5 days 12 hours 45 minutes 48 seconds
  Hardware is 40GigabitEthernet, address is 748e.f8f9.3d80 (bia 748e.f8f9.3d80)
  Configured speed 40Gbit, actual 40Gbit, configured duplex fdx, actual fdx
  Configured mdi mode AUTO, actual none
  Member of 1 L2 VLANs, port is tagged, port state is FORWARDING
  BPDU guard is Disabled, ROOT protect is Disabled, Designated protect is Disabled
  Link Error Dampening is Disabled
  STP configured to ON, priority is level0, mac-learning is enabled
  Flow Control is enabled
  Mirror disabled, Monitor disabled
  Mac-notification is disabled
  Not member of any active trunks
  Not member of any configured trunks
  No port name
  IPG MII 96 bits-time, IPG GMII 96 bits-time
  MTU 1500 bytes, encapsulation ethernet
  Ingress Profile is i1
  Egress Profile is e1
  300 second input rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  300 second output rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  8060797794 packets input, 1031782117647 bytes, 0 no buffer
  Received 0 broadcasts, 0 multicasts, 8060797794 unicasts
  4 input errors, 0 CRC, 0 frame, 0 ignored
  0 runts, 0 giants
  8078157201 packets output, 1034004121728 bytes, 0 underruns
  Transmitted 0 broadcasts, 0 multicasts, 8078157201 unicasts
  0 output errors, 0 collisions
  Relay Agent Information option: Disabled
```

This example shows information for the configured bandwidth on a specific interface. In this example the configured interface bandwidth value is 2000 kilobits.

```
device# show interfaces ethernet 1/1/1

GigabitEthernet1/1/1 is disabled, line protocol is down
  STP Root Guard is disabled, STP BPDU Guard is disabled
  Hardware is GigabitEthernet, address is 748e.f82a.6a00 (bia 748e.f82a.6a00)
  Configured speed auto, actual unknown, configured duplex fdx, actual unknown
Interface bandwidth is 2000 kbps
```

# History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was modified to include PFC status and separate unicast and multicast egress queues. |
| 8.0.30 | This command was modified to include configured bandwidth status. |

# show interfaces lag

Displays information about the LAG interface including counters.

## Syntax

**show interfaces lag** [ *lag-id* | *lag-name* ]

## Parameters

*lag-id*

Displays information for a virtual LAG specified by the LAG ID. If the specified LAG ID is not available, a warning message is displayed.

*lag-name*

Displays information for a virtual LAG specified by the LAG name. If the specified LAG name is not available, a warning message is displayed.

## Modes

Privileged EXEC mode

Global configuration mode

## Examples

The following command shows that the virtual LAG specified by LAG ID 2 is not available in the system.

```
device(config)# show interfaces lag id2
Warning: can't find LAG id2
```

The following command shows information for the virtual LAG named lag1.

```
device# show interfaces lag 1
Lag lg1 is down, line protocol is down
Configured speed Auto, actual None, configured duplex fdx, actual none
Member of L2 VLAN ID 1, port is untagged, port state is None
BPDU guard is Disabled, ROOT protect is Disabled, Designated protect is Disabled
STP configured to ON, priority is level0, mac-learning is enabled
Openflowis Disabled, OpenflowHybrid mode is Disabled
Mirror disabled, Monitor disabled
Mac-notification is disabled
Member of active trunk ports 1/1/10,lg1, Lag Interface is lg1
Member of configured trunk ports 1/1/10,lg1, Lag Interface is lg1
No port name
300 second input rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
300 second output rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
0 packets input, 0 bytes, 0 no buffer
Received 0 broadcasts, 0 multicasts, 0 unicasts
0 input errors, 0 CRC, 0 frame, 0 ignored
0 runts, 0 giants
0 packets output, 0 bytes, 0 underruns
Transmitted 0 broadcasts, 0 multicasts, 0 unicasts
0 output errors, 0 collisions
Relay Agent Information option: Disabled
```

The following command shows information about the virtual LAG interface, including counters.

```
ICX7150-C12 Router(config-lag-blue)#show interfaces lag
Total number of LAGs:          4
Total number of deployed LAGs: 2
Total number of trunks created:2 (126 available)
LACP System Priority / ID:     1 / 609c.9fbc.bf14
LACP Long timeout:             120, default: 120
LACP Short timeout:            3, default: 3
=== LAG "blue" ID 3 (static Not Deployed) ===
LAG Configuration:
   Ports:
   Port Count:    0
   Lag Interface: lg3
   Trunk Type:    hash-based
=== LAG "blue1" ID 10 (dynamic Not Deployed) ===
LAG Configuration:
   Ports:
   Port Count:    0
   Lag Interface: lg10
   Trunk Type:    hash-based
   LACP Key:      20010
=== LAG "test" ID 1 (dynamic Deployed) ===
LAG Configuration:
   Ports:         e 1/1/5 e 1/1/7
   Port Count:    2
   Lag Interface: lg1
   Trunk Type:    hash-based
   LACP Key:      20001
Deployment: HW Trunk ID 1
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/5      Disable None    None None  1     No  1    0   609c.9fbc.bf18

1/1/7      Disable None    None None  1     No  1    0   609c.9fbc.bf18

Port       [Sys P] [Port P] [ Key ] [Act][Tio][Agg][Syn][Col][Dis][Def][Exp][Ope
]
1/1/5           1        1   20001   Yes   S   Agg  Syn  No   No   Def  No   Dwn
1/1/7           1        1   20001   Yes   S   Agg  Syn  No   No   Def  No   Dwn
 Partner Info and PDU Statistics
Port          Partner         Partner     LACP      LACP
            System ID          Key     Rx Count  Tx Count
1/1/5    1-0000.0000.0000        4        0         0
1/1/7    1-0000.0000.0000        6        0         0
 LAG test Counters:
         InOctets                    0          OutOctets                    0
           InPkts                    0            OutPkts                    0
  InBroadcastPkts                    0    OutBroadcastPkts                   0
  InMulticastPkts                    0    OutMulticastPkts                   0
    InUnicastPkts                    0      OutUnicastPkts                   0
        InBadPkts                    0
      InFragments                    0
       InDiscards                    0          OutErrors                    0
              CRC                    0         Collisions                    0
         InErrors                    0      LateCollisions                   0
      InGiantPkts                    0
      InShortPkts                    0
         InJabber                    0
   InFlowCtrlPkts                    0     OutFlowCtrlPkts                   0
     InBitsPerSec                    0       OutBitsPerSec                   0
     InPktsPerSec                    0       OutPktsPerSec                   0
    InUtilization                0.00%      OutUtilization                0.00%
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

| Release version | Command history |
|---|---|
| 08.0.61 | The command was modified to include LAG ID options. |

# show interfaces management

Displays the status of a management interface.

## Syntax

**show interfaces management** [ *mgmt_interface* ]

## Parameters

*mgmt_interface*

Specifies a management interface.

## Modes

Global configuration mode

## Examples

To display the status of a management interface:

```
device(config-vlan-20)# show interfaces management 1
GigEthernetmgmt1 is disabled, line protocol is down
  Port down for 2 minute(s) 26 second(s)
  Hardware is GigEthernet, address is cc4e.24b4.6e64 (bia cc4e.24b4.6e7c)
  Configured speed auto, actual unknown, configured duplex fdx, actual unknown
  Configured mdi mode AUTO, actual unknown
  Member of VLAN 20, port is untagged, port state is NONE
  No port name
  MTU 1500 bytes
  300 second input rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  300 second output rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 multicasts, 0 unicasts
  0 input errors, 0 CRC, 0 frame, 0 ignored
  0 runts, 0 giants
  0 packets output, 0 bytes, 0 underruns
  Transmitted 0 broadcasts, 0 multicasts, 0 unicasts
  0 output errors, 0 collisions
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show interfaces stack-ports

Use the **show interfaces stack-ports** command to display information about the stacking ports for all members in a stack.

## Syntax

**show interfaces stack-ports**

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **clear stack ipc** command before issuing the**show stack ipc** command. This helps to ensure that the data are the most recent traffic statistics for the stack.

This command must be executed from active stack controller.

## Command Output

The **show interfaces stack-ports** command displays the following information:

| Output field | Description |
|---|---|
| Port | Specifies the stack identification number for this unit |
| Link | Identifies the configuration for modules on this unit |
| State | Indicates that a priority has been assigned to this stack unit |
| Dupl | Indicates whether the port is configured as half- or full-duplex |
| Speed | Indicates the port speed |
| Trunk | Indicates whether the port is part of a trunk |
| Tag | Indicates whether the port is tagged or untagged |
| P | Specifies port priority |
| MAC | Provides the MAC address of the port.<br>**NOTE**<br>If a unit is provisional (it is reserved and does not have a physical unit associated with the unit ID), the interface MAC address displayed for the unit is 0000.0000.0000. |
| Name | Displays the optional name assigned to the port if present |

# Examples

The following example displays information about the stack-port interfaces.

```
device# show interfaces stack-ports
Port    Link    State   Dupl Speed Trunk Tag Pvid Pri MAC Name
1/2/1   Up      Forward Full 40G   None  No  N/A  0   748e.f8f9.6315
1/2/2   Down    None    None None  None  No  N/A  0   748e.f8f9.6319
1/2/4   Up      Forward Full 40G   None  No  N/A  0   748e.f8f9.6321
1/2/5   Down    None    None None  None  No  N/A  0   748e.f8f9.6325
1/2/6   Down    None    None None  None  No  N/A  0   748e.f8f9.6329
2/2/1   Up      Forward Full 40G   None  No  N/A  0   cc4e.2438.7295
2/2/4   Up      Forward Full 40G   None  No  N/A  0   cc4e.2438.72a1
2/2/5   Down    None    None None  None  No  N/A  0   cc4e.2438.72a5
3/2/1   Up      Forward Full 40G   None  No  N/A  0   cc4e.2438.7515
3/2/2   Down    None    None None  None  No  N/A  0   cc4e.2438.7519
3/2/3   Down    None    None None  None  No  N/A  0   cc4e.2438.751d
3/2/4   Up      Forward Full 40G   None  No  N/A  0   cc4e.2438.7521
```

# show interfaces tunnel

Displays tunnel interface information.

## Syntax

**show interfaces tunnel** *tunnel-number*

## Parameters

*tunnel-number*

Specifies the tunnel number. Valid values range from 1 through 72.

## Modes

Privileged EXEC mode

## Command Output

The **show interfaces tunnel** command displays the following information:

| Field | Definition |
|---|---|
| Hardware is Tunnel | The interface is a tunnel interface. |
| Tunnel source | The source address for the tunnel. |
| Tunnel destination | The destination address for the tunnel. |
| Tunnel mode | The tunnel mode. The gre specifies that the tunnel will use GRE encapsulation (IP protocol 47). |
| Interface bandwidth | The configured bandwidth on a tunnel interface for routing metric purposes only. |
| Port name | The port name (if applicable). |
| Internet address | The internet address. |
| MTU | The configured path maximum transmission unit. |
| encapsulation GRE | GRE encapsulation is enabled on the port. |
| Keepalive | Indicates whether or not GRE link keepalive is enabled. |
| Path MTU Discovery | Indicates whether or not PMTUD is enabled. If PMTUD is enabled, the MTU value is also displayed. |
| Path MTU | The PMTU that is dynamically learned. |
| Age-timer | Indicates the pmtd aging timer configuration in minutes.The default is 10. The range is from 10 - 30. |
| Path MTU will expire | Indicates the time after which the learned PMTU expires. This line is displayed only when a PMTU is dynamically learned. |

# Examples

This example displays the GRE tunnel configuration and the pmtd aging timer information..

```
show interfaces tunnel 10
Tunnel10 is up, line protocol is up
  Hardware is Tunnel
  Tunnel source  10.1.41.10
  Tunnel destination is 10.1.14.10
  Tunnel mode gre ip
  Port name is GRE_10_to_VR1_on_ICX_STACK
  Internet address is 10.11.1.1/31, MTU 1476 bytes, encapsulation GRE
  Keepalive is not Enabled
  Path MTU Discovery: Enabled, MTU is 1428 bytes, age-timer: 10 minutes
  Path MTU will expire in 0 minutes 50 secs
```

This example shows information for the configured interface bandwidth value on a tunnel interface.

```
device# show interfaces tunnel 2

Tunnel2 is up, line protocol is up
  Hardware is Tunnel
  Tunnel source  10.70.15.1
  Tunnel destination is 10.70.15.2
  Tunnel mode gre ip
  Interface bandwidth is 2000 kbps
  No port name
  Internet address is: 10.0.0.1/24
  Tunnel TOS 0, Tunnel TTL 255, Tunnel MTU 1476 bytes
  Keepalive is not Enabled

  Tunnel Packet Statistics:
                                Unicast Packets                  Multicast Packets
       In-Port(s)          [Rcv-from-tnnl    Xmit-to-tnnl]     [Rcv-from-tnnl    Xmit-to-tnnl]
       e1/1/1 - e1/1/24    2224              0                 0                 0
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was modified to include configured bandwidth status. |

# show interfaces ve

Displays Virtual Ethernet (VE) interface information.

## Syntax

**show interfaces ve** *vlan_id*

## Parameters

*vlan_id*

Specifies the configured corresponding VLAN interface.

## Modes

Privileged EXEC mode

## Examples

This example shows information for the configured bandwidth on a VE interface. In this example the configured interface bandwidth value is 2000 kilobits.

```
device#show interfaces ve 100
Ve100 is up, line protocol is up
  Type is Vlan (Vlan Id: 100)
  Hardware is Virtual Ethernet, address is 748e.f82a.cf00 (bia 748e.f82a.cf00)
  No port name
  Vlan id: 100
  Interface bandwidth is 2000 kbps
 ipv6 address 190::1/64
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was modified to include configured bandwidth status. |

# show ip

Displays global IP configuration information.

## Syntax

**show ip**

## Modes

User EXEC mode

Global configuration mode

## Usage Guidelines

This command has additional options, which are explained in separate command pages.

## Command Output

The **show ip** command displays the following information:

| Field | Description |
|---|---|
| Global settings | |
| ttl | The Time-To-Live (TTL) for IP packets. The TTL specifies the maximum number of router hops a packet can travel before reaching the device. If the packet TTL value is higher than the value specified in this field, the device drops the packet. |
| arp-age | The ARP aging period, which specifies how many minutes an inactive ARP entry remains in the ARP cache before the router ages out the entry. |
| bootp-relay-max-hops | The maximum number of hops away a BootP server can be located from the device and still be used by the router clients for network booting. |
| router-id | The 32-bit number that uniquely identifies the device.<br>By default, the router ID is the numerically lowest IP interface configured on the router. |
| enabled | The IP-related protocols that are enabled on the router. |
| disabled | The IP-related protocols that are disabled on the router. |
| Static routes | |
| Index | The row number of this entry in the IP route table. |
| IP Address | The IP address of the route destination. |
| Subnet Mask | The network mask for the IP address. |
| Next Hop Router | The IP address of the router interface to which the device sends packets for the route. |
| Metric | The cost of the route. Usually, the metric represents the number of hops to the destination. |

| Field | Description |
|---|---|
| Distance | The administrative distance of the route. The default administrative distance for static IP routes in Ruckus ICX devices is 1. |
| Policies | |
| Index | The policy number. This is the number you assigned the policy when you configured it. |
| Action | The action the router takes if a packet matches the comparison values in the policy. The action can be one of the following:<br>• deny: The router drops packets that match this policy.<br>• permit: The router forwards packets that match this policy. |
| Source | The source IP address the policy matches. |
| Destination | The destination IP address the policy matches. |
| Protocol | The IP protocol the policy matches. The protocol can be one of the following:<br>• ICMP<br>• IGMP<br>• IGRP<br>• OSPF<br>• TCP<br>• UDP |
| Port | The Layer 4 TCP or UDP port the policy checks for in packets. The port can be displayed by its number or, for port types the router recognizes, by the well-known name. For example, TCP port 80 can be displayed as HTTP.<br>**NOTE**<br>This field applies only if the IP protocol is TCP or UDP. |
| Operator | The comparison operator for TCP or UDP port names or numbers.<br>**NOTE**<br>This field applies only if the IP protocol is TCP or UDP. |

# Examples

The following example shows sample output of the **show ip** command in which the status of the next hop or ARP port movement syslog message (enabled) is displayed.

```
device(config)# show ip
Global Settings
  ttl: 64, arp-age: 10, bootp-relay-max-hops: 4
  router-id : 10.1.1.1
  enabled : BGP4  UDP-Broadcast-Forwarding  Source-Route  Load-Sharing  RARP  VSRP
  arp-port-move-syslog
  disabled: Route-Only  Directed-Broadcast-Forwarding  IRDP  Proxy-ARP  RIP  OSPF
  VRRP  VRRP-Extended  ICMP-Redirect  add-host-route-first
```

The following example shows sample output of the **show ip** command in which the status of the next hop or ARP port movement syslog message (disabled) is displayed.

```
device(config)# no ip arp port-move-syslog
device(config)# show ip
Global Settings
  ttl: 64, arp-age: 10, bootp-relay-max-hops: 4
  router-id : 10.1.1.1
  enabled : BGP4  UDP-Broadcast-Forwarding  Source-Route  Load-Sharing  RARP  VSRP

  disabled: Route-Only  Directed-Broadcast-Forwarding  IRDP  Proxy-ARP  RIP  OSPF
  VRRP  VRRP-Extended  ICMP-Redirect  add-host-route-first arp-port-move-syslog
```

# History

| Release | Command History |
|---|---|
| 08.0.70 | This command was modified to display the status of the next hop or ARP port movement syslog message(enabled or disabled). |

# show ip access-lists

Displays IPv4 access control list (ACL) information.

## Syntax

**show ip access-lists** [ *acl-num* | *acl-name* ]

## Parameters

*acl-num*

Displays the information for the ACL with the specified ACL number.

*acl-name*

Displays information for the ACL with the specified name.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

ACL configuration mode

## Usage Guidelines

From FastIron release 08.0.50, sequence numbers are automatically added to existing ACL rules, in the following manner:

- The first rule within each ACL is numbered 10.
- The sequence number for each succeeding rule is incremented by 10.

## Examples

The following example displays sample output of the **show ip access-lists** command.

```
device(config-ext-nacl)# show ip access-lists 111
Extended IP access list 111: 4 entries
10: permit ip host 1.1.1.111 host 2.2.2.111
20: permit ospf any any
30: permit pim any any
40: deny ip 20.20.20.96 0.0.0.15 any
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | The command was modified to add sequence numbers automatically to existing rules. |

# show ip arp inspection entries

Displays ARP inspection entries.

## Syntax

**show ip arp inspection entries** { **ethernet** *unit/slot/port* | **ip** *ip-address* | **lag** *lag-id* | **vlan** *vlan-number* }

## Parameters

**ethernet** *unit/slot/port*
: Displays the ARP inspection entries with a specific Ethernet port.

**ip** *ip-address*
: Displays the ARP inspection entries with a specific IP address.

**lag** *lag-id*
: Displays the ARP inspection entries with a specific LAG.

**vlan** *vlan-number*
: Displays the ARP inspection entries with a specific VLAN.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example displays ARP inspection entries with a specific Ethernet port.

```
device# show ip arp inspection entries e 1/3/2
Total Entries: 1
     IP Address    Mac Address      Port   vlan  VRF            Entry Type
     27.0.0.1      cc4e.246e.eb00   1/3/2  2700  default-vrf    arp_entry
```

The following example displays ARP inspection entries with a specific IP address.

```
device# show ip arp inspection entries ip 27.0.0.1
     IP Address  Mac Address      Port    vlan    VRF          Entry Type
     27.0.0.1    cc4e.246e.eb00   1/3/2   2700    default-vrf  arp table entry
```

The following example displays ARP inspection entries with a specific LAG.

```
device# show ip arp inspection entries lag 2
Total Entries: 263
     IP Address      Mac Address    Port   vlan  VRF            Entry Type
     27.1.0.1        0010.2710.04ed lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.2        0010.2710.04ec lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.3        0010.2710.04eb lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.4        0010.2710.04ea lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.5        0010.2710.04e9 lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.6        0010.2710.04e8 lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.7        0010.2710.04f1 lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.8        0010.2710.04f0 lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.9        0010.2710.04ef lg2    2701  default-vrf    dhcp snoop entry
     27.1.0.10       0010.2710.04ee lg2    2701  default-vrf    dhcp snoop entry
```

The following example displays ARP inspection entries with a specific VLAN.

```
device# show ip arp inspection entries vlan 2702
Total Entries: 13
     IP Address      Mac Address    Port  vlan  VRF           Entry Type
     27.2.0.12       0010.2720.04e8 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.13       0010.2720.04e9 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.47       0010.2720.050b lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.48       0010.2720.050c lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.49       0010.2720.050d lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.53       0010.2720.0511 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.54       0010.2720.0512 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.55       0010.2720.0513 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.56       0010.2720.0514 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.57       0010.2720.0515 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.141      0010.2720.0569 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.153      0010.2720.0575 lg2   2702  default-vrf   dhcp snoop entry
     27.2.0.162      0010.2720.057e lg2   2702  default-vrf   dhcp snoop entry
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show ip bgp

Displays entries in the IPv4 Border Gateway Protocol (BGP4) routing table.

## Syntax

**show ip bgp**

**show ip bgp** *ip-addr* [ */prefix* ]

**show ip bgp** *ip-addr* [ */prefix* ] **longer-prefixes**

## Parameters

*ipv6-addr/prefix*

Specifies the IPv4 address and optional prefix.

**longer-prefixes**

Filters on prefixes equal to or greater than that specified by *prefix.*

## Modes

User EXEC mode

## Examples

The following example displays sample output from the **show ip bgp** command.

```
device> show ip bgp
Total number of BGP Routes: 1
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network         Next Hop      Metric  LocPrf   Weight Path
*> 10.1.1.0/24    192.168.1.5    1       100      0      90000 100 200 65535
65536 65537 65538 65539 75000
```

The following example displays sample output from the **show ip bgp** command when an IP address is specified.

```
device> show ip bgp 10.3.4.0

Number of BGP Routes matching display condition : 1
Status codes: s suppressed, d damped, h history, * valid, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop        Metric LocPrf Weight Path
*>  10.3.4.0/24         192.168.4.106          100    0      65001 4355 1 1221 ?
       Last update to IP routing table: 0h11m38s, 1 path(s) installed:
         Gateway           Port
         192.168.2.1       1/2/1
       Route is advertised to 1 peers:
        10.20.20.2(65300)
```

# show ip bgp attribute-entries

Displays BGP4 route-attribute entries that are stored in device memory.

## Syntax

**show ip bgp attribute-entries**

## Modes

User EXEC mode

## Usage Guidelines

The route-attribute entries table lists the sets of BGP4 attributes that are stored in device memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the device typically has fewer attribute entries than routes. Use this command to view BGP4 route-attribute entries that are stored in device memory.

## Command Output

The **show ip bgp attribute-entries** command displays the following information:

| Output field | Description |
|---|---|
| Total number of BGP4 Attribute Entries | The number of routes contained in this BGP4 route table. |
| Next Hop | The IP address of the next-hop device for routes that have this set of attributes. |
| Metric | The cost of the routes that have this set of attributes. |
| Origin | The source of the route information. The origin can be one of the following:<br>• EGP - The routes with these attributes came to BGP4 through EGP.<br>• IGP - The routes with these attributes came to BGP4 through IGP.<br>• INCOMPLETE - The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP.<br>When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE. |
| Originator | The originator of the route in a route reflector environment. |
| Cluster List | The route-reflector clusters through which this set of attributes has passed. |
| Aggregator | Aggregator information:<br>• AS Number shows the AS in which the network information in the attribute set was aggregated. This value applies only to aggregated routes and is otherwise 0.<br>• Router-ID shows the device that originated this aggregator. |

| Output field | Description |
|---|---|
| Atomic | Whether the network information in this set of attributes has been aggregated *and* this aggregation has resulted in information loss.<br>• TRUE - Indicates information loss has occurred<br>• FALSE - Indicates no information loss has occurred<br>**NOTE**<br>Information loss under these circumstances is a normal part of BGP4 and does not indicate an error. |
| Local Pref | The degree of preference for routes that use these attributes relative to other routes in the local AS. |
| Communities | The communities to which routes with these attributes belong. |
| AS Path | The autonomous systems through which routes with these attributes have passed. The local AS is shown in parentheses. |

# Examples

The following example show sample output for the **show ip bgp attribute-entries** command.

```
device> show ip bgp attribute-entries

        Total number of BGP Attribute Entries: 18 (0)
1       Next Hop  :192.168.1.6        MED :1             Origin:INCOMP
        Originator:0.0.0.0            Cluster List:None
        Aggregator:AS Number :0      Router-ID:0.0.0.0     Atomic:None
        Local Pref:100               Communities:Internet
AS Path   :90000 80000 (length 11
)
        Address: 0x10e4e0c4  Hash:489 (0x03028536), PeerIdx 0
        Links: 0x00000000, 0x00000000, nlri: 0x10f4804a
        Reference Counts: 1:0:1, Magic: 51
2       Next Hop  :192.168.1.5        Metric   :1            Origin:INCOMP
        Originator:0.0.0.0            Cluster List:None
        Aggregator:AS Number :0      Router-ID:0.0.0.0       Atomic:None
        Local Pref:100               Communities:Internet
AS Path   :90000 75000 (length 11)
        Address: 0x10e4e062  Hash:545 (0x0301e8f6), PeerIdx 0
        Links: 0x00000000, 0x00000000, nlri: 0x10f47ff0
        Reference Counts: 1:0:1, Magic: 49
```

# show ip bgp config

Displays active BGP4 configuration information.

## Syntax

**show ip bgp config**

## Modes

User EXEC mode

## Examples

The following example displays sample output from the **show ip bgp config** command.

```
device> show ip bgp config

router bgp
  local-as 200
neighbor 10.102.1.1 remote-as 200
neighbor 10.102.1.1 ebgp-multihop
neighbor 10.102.1.1 update-source loopback 1
neighbor 192.168.2.1 remote-as 100
neighbor 10.200.2.2 remote-as 400
neighbor 2001:db8::1:1 remote-as 200
neighbor 2001:db8::1:2 remote-as 400
neighbor 2001:db8::1 remote-as 300

address-family ipv4 unicast
no neighbor 2001:db8::1:1 activate
no neighbor 2001:db8::1:2 activate
no neighbor 2001:db8::1 activate
exit-address-family

address-family ipv6 unicast
redistribute static
neighbor 2001:db8::1:1 activate
neighbor 2001:db8::1:2 activate
neighbor 2001:db8::1 activate
exit-address-family
end of BGP configuration
```

# show ip bgp dampened-paths

Displays all BGP4 dampened routes.

## Syntax

**show ip bgp dampened-paths**

## Modes

User EXEC mode

# show ip bgp filtered-routes

Displays BGP4 filtered routes that are received from a neighbor or peer group.

## Syntax

**show ip bgp filtered-routes** [ **detail** ] [ *ip-addr* { **/** *mask* } [ **longer-prefixes** ] | **as-path-access-list** *name* | **prefix-list** *name* ]

## Parameters

**detail**

Displays detailed route information.

*ip-addr*

Specifies the IPv4 address of the destination network in dotted-decimal notation.

*mask*

Specifies the IPv4 mask of the destination network in CIDR notation.

**longer-prefixes**

Specifies all statistics for routes that match the specified route, or that have a longer prefix than the specified route.

**as-path-access-list** *name*

Specifies an AS-path ACL. The name must be between 1 and 32 ASCII characters in length.

**prefix-list** *name*

Specifies an IP prefix list. The name must be between 1 and 32 ASCII characters in length.

*name*

Specifies the name of an AS-path ACL or prefix list.

## Modes

User EXEC mode

## Examples

The following example displays BGP4 filtered routes.

```
device> show ip bgp filtered-routes
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix             Next Hop        MED       LocPrf     Weight Status
1      10.3.0.0/8          192.168.4.106             100        0      EF
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106             100        0      EF
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106             100        0      EF
         AS_PATH: 65001 4355 701 1 189
```

# show ip bgp flap-statistics

Displays BGP4 route-dampening statistics for all dampened routes with a variety of options.

## Syntax

**show ip bgp flap-statistics**

**show ip bgp flap-statistics** *ip-addr* { **/** *mask* } [ **longer-prefix** ]

**show ip bgp flap-statistics as-path-filter** *name*

**show ip bgp flap-statistics neighbor** *ip-addr*

**show ip bgp flap-statistics regular-expression** *name*

## Parameters

*ip-addr*
IPv4 address of a specified route in dotted-decimal notation.

*mask*
IPv4 mask of a specified route in CIDR notation.

**longer-prefixes**
Displays statistics for routes that match the specified route or have a longer prefix than the specified route.

**as-path-filter** *name*
Specifies an AS-path filter.

**neighbor**
Displays flap statistics only for routes learned from the specified neighbor.

*ip-addr*
IPv4 address of the neighbor.

**regular-expression**
Specifies a regular expression in the display output on which to filter.

*name*
Name of an AS-path filter or regular expression.

## Modes

User EXEC mode

## Command Output

The **show ip bgp flap-statistics** command displays the following information:

| Output field | Description |
|---|---|
| Total number of flapping routes | The total number of routes in the BGP4 route table that have changed state and have been marked as flapping routes. |

| Output field | Description |
|---|---|
| Status code | Indicates the dampening status of the route, which can be one of the following:<br>• > - This is the best route among those in the BGP4 route table to the route destination.<br>• d - This route is currently dampened, and unusable.<br>• h - The route has a history of flapping and is unreachable now.<br>• * - The route has a history of flapping but is currently usable. |
| Network | The destination network of the route. |
| From | The neighbor that sent the route to the device. |
| Flaps | The number of flaps the route has experienced. |
| Since | The amount of time since the first flap of this route. |
| Reuse | The amount of time remaining until this route will be un-suppressed and can be used again. |
| Path | Shows the AS-path information for the route. |

# Examples

The following example displays route dampening statistics.

```
device> show ip bgp flap-statistics
Total number of flapping routes: 414
    Status Code  >:best d:damped h:history *:valid
    Network            From         Flaps Since    Reuse    Path
h>  10.50.206.0/23   10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  10.255.192.0/20  10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  10.252.165.0/24  10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  10.50.208.0/23   10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  10.33.0.0/16     10.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
*>  10.17.220.0/24   10.90.213.77   1     0 :1 :4  0 :0 :0  65001 4355 701 62
```

# show ip bgp ipv6

Displays IPv6 unicast information.

## Syntax

**show ip bgp ipv6 neighbors**

**show ip bgp ipv6 neighbors** *ip-addr* **advertised-routes** [ **detail** ] [ *ipv6 address /mask* ]

**show ip bgp ipv6 neighbors** *ip-addr* **flap-statistics**

**show ip bgp ipv6 neighbors** *ip-addr* **last-packet-with-error** [ **decode** ]

**show ip bgp ipv6 neighbors** *ip-addr* **received** [ **prefix-filter** ]

**show ip bgp ipv6 neighbors** *ip-addr* **received-routes** [ **detail** ]

**show ip bgp ipv6 neighbors** *ip-addr* **rib-out-routes** [ **detail** ] [ *ipv6 address /mask* ]

**show ip bgp ipv6 neighbors** *ip-addr* **routes**

**show ip bgp ipv6 neighbors** *ip-addr* **routes** { **best** | **not-installed-best** | **unreachable** }

**show ip bgp ipv6 neighbors** *ip-addr* **routes detail** { **best** | **not-installed-best** | **unreachable** }

**show ip bgp ipv6 neighbors** *ip-addr* **routes-summary**

**show ip bgp ipv6 neighbors last-packet-with-error**

**show ip bgp ipv6 neighbors routes-summary**

**show ip bgp ipv6 summary**

## Parameters

**neighbors**
Specifies a neighbor.

*ip-addr*
IPv4 address of a neighbor in dotted-decimal notation.

**advertised-routes**
Specifies the routes that the device has advertised to the neighbor during the current BGP4 session.

**detail**
Specifies detailed information.

*ipv6 address /mask*
Specifies an IPv6 address and mask.

**flap-statistics**
Specifies the route flap statistics for routes received from or sent to a BGP4 neighbor.

**last-packet-with-error**
Specifies the last packet with an error.

**decode**
Decodes the last packet that contained an error from any of a device's neighbors.

**received**
Specifies Outbound Route Filters (ORFs) received from BGP4 neighbors of the device.

**prefix-filter**
Displays the results for ORFs that are prefix-based.

**received-routes**
Specifies all route information received in route updates from BGP4 neighbors of the device since the soft-reconfiguration feature was enabled.

**rib-out-routes**
Displays information about the current BGP4 Routing Information Base (Adj-RIB-Out) for specific neighbors and specific destination networks.

**routes**
Displays a variety of route information received in UPDATE messages from BGP4 neighbors.

**best**
Displays routes received from the neighbor that are the best BGP4 routes to their destination.

**not-installed-best**
Displays routes received from the neighbor that are the best BGP4 routes to their destination but were not installed in the route table because the device received better routes from other sources.

**unreachable**
Displays routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.

**routes-summary**
Displays all route information received in UPDATE messages from BGP4 neighbors.

**summary**
Displays summarized IPv6 unicast information.

# Modes

User EXEC mode

# Examples

The following example displays summarized IPv6 unicast information.

```
device> show ip bgp ipv6 summary
BGP4 Summary
Router ID: 10.1.1.1 Local AS Number: 1
Confederation Identifier: not configured
Confederation Peers:
Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
Number of Neighbors Configured: 1, UP: 1
Number of Routes Installed: 1, Uses 86 bytes
Number of Routes Advertising to All Neighbors: 0 (0 entries)
Number of Attribute Entries Installed: 1, Uses 90 bytes
Neighbor Address AS# State Time Rt:Accepted Filtered Sent ToSend
192.168.1.2 2 ESTAB 0h 1m51s 1 0 0 0
```

The following example displays IPv6 unicast device information with respect to IPv4 neighbors.

```
device(config-bgp)# show ip bgp ipv6 neighbors
Total number of BGP Neighbors: 1
1 IP Address: 192.168.1.2, AS: 2 (EBGP), RouterID: 10.1.1.2, VRF: default-vrf
State: ESTABLISHED, Time: 0h8m33s, KeepAliveTime: 60, HoldTime: 180
KeepAliveTimer Expire in 17 seconds, HoldTimer Expire in 135 seconds
UpdateSource: Loopback 1
RefreshCapability: Received
.......
Neighbor NLRI Negotiation:
Peer Negotiated IPV6 unicast capability
Peer configured for IPV6 unicast Routes
Neighbor AS4 Capability Negotiation:
TCP Connection state: ESTABLISHED, flags:00000033 (0,0)
```

# show ip bgp neighbors

Displays configuration information and statistics for BGP4 neighbors of the device.

## Syntax

**show ip bgp neighbors** [ *ip-addr* ]

**show ip bgp neighbors last-packet-with-error**

**show ip bgp neighbors routes-summary**

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**

Displays the last packet with an error.

**routes-summary**

Displays routes received, routes accepted, number of routes advertised by peer, and so on.

## Modes

User EXEC mode

## Usage Guidelines

Output shows all configured parameters for the neighbors. Only the parameters whose values differ from defaults are shown.

## Command Output

The **show ip bgp neighbors** command displays the following information:

| Output field | Description |
|---|---|
| Total Number of BGP4 Neighbors | The number of BGP4 neighbors configured. |
| IP Address | The IP address of the neighbor. |
| AS | The AS the neighbor is in. |
| EBGP or IBGP | Whether the neighbor session is an IBGP session, an EBGP session, or a confederation EBGP session:<br>• EBGP - The neighbor is in another AS.<br>• EBGP_Confed - The neighbor is a member of another sub-AS in the same confederation.<br>• IBGP - The neighbor is in the same AS. |
| RouterID | The neighbor device ID. |
| Description | The description you gave the neighbor when you configured it on the device. |

| Output field | Description |
|---|---|
| Local AS | The value (if any) of the Local AS configured. |
| State | The state of the session with the neighbor. The states are from the device perspective, not the neighbor perspective. The state values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each device:<br>• IDLE - The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• ADMND - The neighbor has been administratively shut down. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• CONNECT - BGP4 is waiting for the connection process for the TCP neighbor session to be completed.<br>• ACTIVE - BGP4 is waiting for a TCP connection from the neighbor.<br>**NOTE**<br>If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.<br>• OPEN SENT - BGP4 is waiting for an Open message from the neighbor.<br>• OPEN CONFIRM - BGP4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the device receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.<br>• ESTABLISHED - BGP4 is ready to exchange UPDATE messages with the neighbor.<br>If there is more BGP4 data in the TCP receiver queue, a plus sign (+) is also displayed.<br>**NOTE**<br>If you display information for the neighbor using the **show ip bgp neighbor***ip-addr* command, the TCP receiver queue value will be greater than 0. |
| Time | The amount of time this session has been in the current state. |
| KeepAliveTime | The keep alive time, which specifies how often this device sends keepalive messages to the neighbor. |
| HoldTime | The hold time, which specifies how many seconds the device will wait for a keepalive or update message from a BGP4 neighbor before deciding that the neighbor is not operational. |
| PeerGroup | The name of the peer group the neighbor is in, if applicable. |
| Multihop-EBGP | Whether this option is enabled for the neighbor. |
| RouteReflectorClient | Whether this option is enabled for the neighbor. |
| SendCommunity | Whether this option is enabled for the neighbor. |
| NextHopSelf | Whether this option is enabled for the neighbor. |
| DefaultOriginate | Whether this option is enabled for the neighbor. |

| Output field | Description |
|---|---|
| MaximumPrefixLimit | Maximum number of prefixes the device will accept from this neighbor. |
| RemovePrivateAs | Whether this option is enabled for the neighbor. |
| RefreshCapability | Whether this device has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability. |
| CooperativeFilteringCapability | Whether the neighbor is enabled for cooperative route filtering. |
| Distribute-list | Lists the distribute list parameters, if configured. |
| Filter-list | Lists the filter list parameters, if configured. |
| Prefix-list | Lists the prefix list parameters, if configured. |
| Route-map | Lists the route map parameters, if configured. |
| Messages Sent | The number of messages this device has sent to the neighbor. The display shows statistics for the following message types:<br>• Open<br>• Update<br>• KeepAlive<br>• Notification<br>• Refresh-Req |
| Messages Received | The number of messages this device has received from the neighbor. The message types are the same as for the Message Sent field. |
| Last Update Time | Lists the last time updates were sent and received for the following:<br>• NLRIs<br>• Withdraws |
| Last Connection Reset Reason | The reason the previous session with this neighbor ended. The reason can be one of the following:<br>Reasons described in the BGP4 specifications:<br>• Message Header Error<br>• Connection Not Synchronized<br>• Bad Message Length<br>• Bad Message Type<br>• OPEN Message Error<br>• Unsupported Version Number<br>• Bad Peer AS Number<br>• Bad BGP4 Identifier<br>• Unsupported Optional Parameter<br>• Authentication Failure<br>• Unacceptable Hold Time<br>• Unsupported Capability<br>• UPDATE Message Error<br>• Malformed Attribute List<br>• Unrecognized Well-known Attribute<br>• Missing Well-known Attribute<br>• Attribute Flags Error<br>• Attribute Length Error<br>• Invalid ORIGIN Attribute<br>• Invalid NEXT_HOP Attribute |

| Output field | Description |
|---|---|
| | • Optional Attribute Error<br>• Invalid Network Field<br>• Malformed AS_PATH<br>• Hold Timer Expired<br>• Finite State Machine Error<br>• Rcv Notification |
| Last Connection Reset Reason (cont.) | Reasons specific to the Ruckus implementation:<br>• Reset All Peer Sessions<br>• User Reset Peer Session<br>• Port State Down<br>• Peer Removed<br>• Peer Shutdown<br>• Peer AS Number Change<br>• Peer AS Confederation Change<br>• TCP Connection KeepAlive Timeout<br>• TCP Connection Closed by Remote<br>• TCP Data Stream Error Detected |
| Notification Sent | If the device receives a NOTIFICATION message from the neighbor, the message contains an error code corresponding to one of the following errors. Some errors have subcodes that clarify the reason for the error. Where applicable, the subcode messages are listed underneath the error code messages.<br>• Message Header Error:<br>– Connection Not Synchronized<br>– Bad Message Length<br>– Bad Message Type<br>– Unspecified<br>• Open Message Error:<br>– Unsupported Version<br>– Bad Peer As<br>– Bad BGP4 Identifier<br>– Unsupported Optional Parameter<br>– Authentication Failure<br>– Unacceptable Hold Time<br>– Unspecified<br>• Update Message Error:<br>– Malformed Attribute List<br>– Unrecognized Attribute<br>– Missing Attribute<br>– Attribute Flag Error<br>– Attribute Length Error<br>– Invalid Origin Attribute<br>– Invalid NextHop Attribute<br>– Optional Attribute Error<br>– Invalid Network Field<br>– Malformed AS Path<br>– Unspecified<br>• Hold Timer Expired<br>• Finite State Machine Error<br>• Cease<br>• Unspecified |
| Notification Received | Refer to details for the field Notification Sent. |

| Output field | Description |
|---|---|
| TCP Connection state | The state of the connection with the neighbor. The connection can have one of the following states:<br>• LISTEN - Waiting for a connection request.<br>• SYN-SENT - Waiting for a matching connection request after having sent a connection request.<br>• SYN-RECEIVED - Waiting for a confirming connection request acknowledgment after having both received and sent a connection request.<br>• ESTABLISHED - Data can be sent and received over the connection. This is the normal operational state of the connection.<br>• FIN-WAIT-1 - Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent.<br>• FIN-WAIT-2 - Waiting for a connection termination request from the remote TCP.<br>• CLOSE-WAIT - Waiting for a connection termination request from the local user.<br>• CLOSING - Waiting for a connection termination request acknowledgment from the remote TCP.<br>• LAST-ACK - Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request).<br>• TIME-WAIT - Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request.<br>• CLOSED - There is no connection state. |
| Byte Sent | The number of bytes sent. |
| Byte Received | The number of bytes received. |
| Local host | The IP address of the device. |
| Local port | The TCP port the device is using for the BGP4 TCP session with the neighbor. |
| Remote host | The IP address of the neighbor. |
| Remote port | The TCP port the neighbor is using for the BGP4 TCP session with the device. |
| ISentSeq | The initial send sequence number for the session. |
| SendNext | The next sequence number to be sent. |
| TotUnAck | The number of sequence numbers sent by the device that have not been acknowledged by the neighbor. |
| TotSent | The number of sequence numbers sent to the neighbor. |
| ReTrans | The number of sequence numbers that the device retransmitted because they were not acknowledged. |
| UnAckSeq | The current acknowledged sequence number. |
| IRcvSeq | The initial receive sequence number for the session. |
| RcvNext | The next sequence number expected from the neighbor. |
| SendWnd | The size of the send window. |
| TotalRcv | The number of sequence numbers received from the neighbor. |
| DupliRcv | The number of duplicate sequence numbers received from the neighbor. |

| Output field | Description |
|---|---|
| RcvWnd | The size of the receive window. |
| SendQue | The number of sequence numbers in the send queue. |
| RcvQue | The number of sequence numbers in the receive queue. |
| CngstWnd | The number of times the window has changed. |

# Examples

The following example shows sample output from the show ip bgp neighbors command.

```
device> show ip bgp neighbors
  neighbors              Details on TCP and BGP neighbor connections
    Total number of BGP Neighbors: 1
1   IP Address: 192.168.1.1, AS: 7701000 (IBGP), RouterID: 192.168.1.1, VRF: default-vrf
    State: ESTABLISHED, Time: 0h3m33s, KeepAliveTime: 60, HoldTime: 180
       KeepAliveTimer Expire in 49 seconds, HoldTimer Expire in 177 seconds
    Minimal Route Advertisement Interval: 0 seconds
       RefreshCapability: Received
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 1       0       5         0            0
       Received: 1       1       5         0            0
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                  Tx: ---        ---           Rx: 0h3m33s    ---
    Last Connection Reset Reason:Unknown
    Notification Sent:     Unspecified
    Notification Received: Unspecified
    Neighbor NLRI Negotiation:
      Peer Negotiated IPV4  unicast  capability
      Peer configured for IPV4 unicast  Routes
    Neighbor AS4 Capability Negotiation:
      Peer Negotiated AS4  capability
      Peer configured for AS4  capability

 As-path attribute count: 1
    Outbound Policy Group:
       ID: 1, Use Count: 1
    TCP Connection state: ESTABLISHED, flags:00000044 (0,0)
     Maximum segment size: 1460
    TTL check: 0, value: 0, rcvd: 64
       Byte Sent:   148, Received: 203
       Local host:  192.168.1.2, Local  Port: 179
       Remote host: 192.168.1.1, Remote Port: 8041
       ISentSeq:    1656867  SendNext:    1657016  TotUnAck:          0
       TotSent:         149  ReTrans:          19  UnAckSeq:    1657016
       IRcvSeq:     1984547  RcvNext:     1984751  SendWnd:       64981
       TotalRcv:        204  DupliRcv:        313  RcvWnd:        65000
       SendQue:           0  RcvQue:            0  CngstWnd:       5840
```

# show ip bgp neighbors advertised-routes

Displays the routes that the device has advertised to the neighbor during the current BGP4 session.

## Syntax

**show ip bgp neighbors** *ip-addr* **advertised-routes** [ **detail** | **/** *mask-bits* ]

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

**detail**

Specifies detailed information.

*mask-bits*

Specifies the number of mask bits in CIDR notation.

## Modes

User EXEC mode

## Examples

The following example displays the routes the device has advertised to a specified neighbor.

```
device> show ip bgp neighbors 192.168.4.211 advertised-routes

       There are 2 routes advertised to neighbor 192.168.4.211
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Network        Next Hop       Metric   LocPrf   Weight   Status
1      10.102.0.0/24  192.168.2.102   12               32768    BL
2      10.200.1.0/24  192.168.2.102   0                32768    BL
```

# show ip bgp neighbors flap-statistics

Displays the route flap statistics for routes received from or sent to a BGP4 neighbor.

## Syntax

**show ip bgp neighbors** *ip-addr* **flap-statistics**

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

## Modes

User EXEC mode

# show ip bgp neighbors last-packet-with-error

Displays the last packets with an error from BGP4 neighbors of the device.

## Syntax

**show ip bgp neighbors** *ip-addr* **last-packet-with-error** [ **decode** ]

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**decode**

Decodes the last packet that contained an error from any of a device's neighbors.

## Modes

User EXEC mode

# show ip bgp neighbors received

Displays Outbound Route Filters (ORFs) received from BGP4 neighbors of the device.

## Syntax

**show ip bgp neighbors** *ip-addr* **received** { **extended-community** | **prefix-filter** }

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

**extended-community**

Displays the results for ORFs that use the BGP Extended Community Attribute.

**prefix-filter**

Displays the results for ORFs that are prefix-based.

## Modes

User EXEC mode

## Examples

The following example displays sample output for the **show ip bgp neighbors received** command when the **prefix-filter** keyword is used.

```
device> show ip bgp neighbor 10.10.10.1 received prefix-filter

ip prefix-list 10.10.10.1: 4 entries
     seq 5 permit 10.10.0.0/16 ge 18 le 28
     seq 10 permit 10.20.10.0/24
     seq 15 permit 10.0.0.0/8 le 32
     seq 20 permit 10.10.0.0/16 ge 18
```

# show ip bgp neighbors received-routes

Lists all route information received in route updates from BGP4 neighbors of the device since the soft-reconfiguration feature was enabled.

## Syntax

**show ip bgp neighbors** *ip-addr* **received-routes** [ **detail** ]

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

**detail**

Displays detailed route information.

## Modes

User EXEC mode

## Examples

The following example displays the details of route updates.

```
device> show ip bgp neighbor 10.168.4.106 received-routes

       There are 97345 received routes from  neighbor 10.168.4.106
Searching for matching routes, use ^C to quit...
tatus A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTEREDtatus A:AGGREGATE B:BEST
b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix              Next Hop        MED     LocPrf    Weight Status
1      10.3.0.0/8          10.168.4.106            100       0      BE
         AS_PATH: 65001 4355 701 8
2      10.4.0.0/8          10.168.4.106            100       0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      10.168.4.106            100       0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          10.168.4.106            100       0      BE
```

# show ip bgp neighbors rib-out-routes

Displays information about the current BGP4 Routing Information Base (Adj-RIB-Out) for specific neighbors and specific destination networks.

## Syntax

**show ip bgp neighbors** *ip-addr* **rib-out-routes** [ **detail** ] [*ip-addr* [ **/** *mask* ] ]

## Parameters

*ip-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**

Displays the last packet with an error.

**routes-summary**

Displays routes received, routes accepted, number of routes advertised by peer, and so on.

## Modes

User EXEC mode

## Examples

The following example shows information about the routes that the device either has most recently sent, or is about to send, to a specified neighbor and a specified destination network

```
device> show ip bgp neighbor 192.168.4.211 rib-out-routes 192.168.1.0/24

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      10.200.1.0/24       0.0.0.0         0          101         32768  BL
```

# show ip bgp neighbors routes

Lists a variety of route information received in UPDATE messages from BGP4 neighbors.

## Syntax

**show ip bgp neighbors** *ip-addr* **routes**

**show ip bgp neighbors** *ip-addr* **routes** { **best** | **not-installed-best** | **unreachable** }

**show ip bgp neighbors** *ip-addr* **routes detail** { **best** | **not-installed-best** | **unreachable** }

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

**best**

Displays routes received from the neighbor that are the best BGP4 routes to their destination.

**not-installed-best**

Displays routes received from the neighbor that are the best BGP4 routes to their destination but were not installed in the route table because the device received better routes from other sources.

**unreachable**

Displays routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.

## Modes

User EXEC mode

## Examples

The following example shows sample output for the **show ip bgp neighbors routes** command.

```
device> show ip bgp neighbors 192.168.4.106 routes

       There are 97345 received routes from  neighbor 192.168.4.106
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix             Next Hop        MED    LocPrf    Weight Status
1      10.3.0.0/8          192.168.4.106          100       0      BE
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106          100       0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106          100       0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          192.168.4.106                    0      BE
…
```

# show ip bgp neighbors routes-summary

Lists all route information received in UPDATE messages from BGP4 neighbors.

## Syntax

**show ip bgp neighbors** *ip-addr* **routes-summary**

## Parameters

*ip-addr*

IPv4 address of a neighbor in dotted-decimal notation.

## Modes

User EXEC mode

## Command Output

The **show ip bgp neighbors routes-summary** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IP address of the neighbor. |
| Routes Received | How many routes the device has received from the neighbor during the current BGP4 session:<br>• Accepted or Installed - Number of received routes the device accepted and installed in the BGP4 route table.<br>• Filtered or Kept - Number of routes that were filtered out, but were retained in memory for use by the soft reconfiguration feature.<br>• Filtered - Number of received routes filtered out. |
| Routes Selected as BEST Routes | The number of routes that the device selected as the best routes to their destinations. |
| BEST Routes not Installed in IP Forwarding Table | The number of routes received from the neighbor that are the best BGP4 routes to their destinations, but were not installed in the IP route table because the device received better routes from other sources (such as OSPF, RIP, or static IP routes). |
| Unreachable Routes | The number of routes received from the neighbor that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next-hop. |
| History Routes | The number of routes that are down but are being retained for route flap dampening purposes. |
| NLRIs Received in Update Message | The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages:<br>• Withdraws - Number of withdrawn routes the device has received.<br>• Replacements - Number of replacement routes the device has received. |

| Output field | Description |
|---|---|
| NLRIs Discarded due to | Indicates the number of times the device discarded an NLRI for the neighbor due to the following reasons:<br>• Maximum Prefix Limit - The configured maximum prefix amount had been reached.<br>• AS Loop - An AS loop occurred. An AS loop occurs when the BGP4 AS-path attribute contains the local AS number.<br>• maxas-limit aspath - The number of route entries discarded because the AS path exceeded the configured maximum length or exceeded the internal memory limits.<br>• Invalid Nexthop - The next-hop value was not acceptable.<br>• Duplicated Originator_ID - The originator ID was the same as the local device ID.<br>• Cluster_ID - The cluster list contained the local cluster ID, or the local device ID if the cluster ID is not configured. |
| Routes Advertised | The number of routes the device has advertised to this neighbor:<br>• To be Sent - The number of routes queued to send to this neighbor.<br>• To be Withdrawn - The number of NLRIs for withdrawing routes the device has queued to send to this neighbor in UPDATE messages. |
| NLRIs Sent in Update Message | The number of NLRIs for new routes the device has sent to this neighbor in UPDATE messages:<br>• Withdraws - Number of routes the device has sent to the neighbor to withdraw.<br>• Replacements - Number of routes the device has sent to the neighbor to replace routes the neighbor already has. |
| Peer Out of Memory Count for | Statistics for the times the device has run out of BGP4 memory for the neighbor during the current BGP4 session:<br>• Receiving Update Messages - The number of times UPDATE messages were discarded because there was no memory for attribute entries.<br>• Accepting Routes (NLRI) - The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count.<br>• Attributes - The number of times there was no memory for BGP4 attribute entries.<br>• Outbound Routes (RIB-out) - The number of times there was no memory to place a "best" route into the neighbor route information base (Adj-RIB-Out) for routes to be advertised. |

# Examples

The following example displays route summary information received in UPDATE messages.

```
device> show ip bgp neighbor 10.168.4.211 routes-summary

1   IP Address: 10.168.4.211
Routes Accepted/Installed:1,  Filtered/Kept:11,  Filtered:11
   Routes Selected as BEST Routes:1
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0

NLRIs Received in Update Message:24,  Withdraws:0 (0),  Replacements:1
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
      Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:0,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:0,  Withdraws:0,  Replacements:0

Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
   Attributes:0, Outbound Routes(RIB-out):0
```

# show ip bgp peer-group

Displays peer-group information.

## Syntax

**show ip bgp peer-group** *peer-group-name*

## Parameters

*peer-group-name*
Specifies a peer group name.

## Modes

User EXEC mode

## Usage Guidelines

Only the parameters that have values different from their defaults are listed.

## Examples

The following example shows sample output from the **show ip bgp peer-group** command.

```
device> show ip bgp peer-group pg1
1   BGP peer-group is pg
    Description: peer group abc
       SendCommunity: yes
       NextHopSelf: yes
       DefaultOriginate: yes
    Members:
       IP Address: 10.168.10.10, AS: 65111
```

# show ip bgp routes

Displays statistics for the routes in the BGP4 route table of a device.

## Syntax

**show ip bgp routes** [ **detail** ] [ *num* | *ip-address/prefix* | **age** *num* | **as-path-access-list** *name* | **as-path-filter** *number* | **best** | **cidr-only** | **community-access-list** *name* | **community-filter** *number* | **community-reg-expression** *expression* | **local** | **neighbor** *ip-addr* | **nexthop** *ip-addr* | **no-best** | **not-installed-best** | **prefix-list** *string* | **regular-expression** *name* | **route-map** *name* | **summary** | **unreachable** ]

## Parameters

**detail**
Displays detailed information.

*num*
Table entry at which the display starts. For example, if you want to list entries beginning with table entry 100, specify 100.

*ip-address/prefix*
Specifies an IP address and prefix.

**age** *num*
Displays BGP4 route information that is filtered by age.

**as-path-access-list** *name*
Displays BGP4 route information that is filtered by autonomous system (AS)-path access control list (ACL).

**as-path-filter** *number*
Displays BGP4 route information that is filtered using the specified AS-path filter.

**best**
Displays BGP4 route information that the device selected as best routes.

**cidr-only**
Displays BGP4 routes whose network masks do not match their class network length.

**community-access-list** *name*
Displays BGP4 route information for an AS-path community access list.

**community-filter** *number*
Displays BGP4 route information that matches a specific community filter.

**community-reg-expression** *expression*
Displays BGP4 route information for an ordered community list regular expression.

**local**
Displays BGP4 route information about selected local routes.

**neighbor** *ip-addr*
Displays BGP4 route information about selected BGP neighbors.

**nexthop** *ip-addr*
Displays BGP4 route information about routes that are received from the specified next hop.

**no-best**
Displays BGP4 route information that the device selected as not best routes.

**not-installed-best**
Displays BGP4 route information about best routes that are not installed.

**prefix-list** *string*
Displays BGP4 route information that is filtered by a prefix list.

**regular-expression** *name*
Displays BGP4 route information about routes that are associated with the specified regular expression.

**route-map** *name*
Displays BGP4 route information about routes that use the specified route map.

**summary**
Displays BGP4 summary route information.

**unreachable**
Displays BGP4 route information about routes whose destinations are unreachable through any of the BGP4 paths in the BGP4 route table.

# Modes

User EXEC mode

# Command Output

The **show ip bgp routes** command displays the following information:

| Output field | Description |
|---|---|
| Total number of BGP4 routes (NLRIs) Installed | Number of BGP4 routes the device has installed in the BGP4 route table. |
| Distinct BGP4 destination networks | Number of destination networks the installed routes represent. The BGP4 route table can have multiple routes to the same network. |
| Filtered BGP4 routes for soft reconfig | Number of route updates received from soft-reconfigured neighbors or peer groups that have been filtered out but retained. |
| Routes originated by this device | Number of routes in the BGP4 route table that this device originated. |
| Routes selected as BEST routes | Number of routes in the BGP4 route table that this device has selected as the best routes to the destinations. |
| BEST routes not installed in IP forwarding table | Number of BGP4 routes that are the best BGP4 routes to their destinations but were not installed in the IP route table because the device received better routes from other sources (such as OSPF, RIP, or static IP routes). |
| Unreachable routes (no IGP route for NEXTHOP) | Number of routes in the BGP4 route table whose destinations are unreachable because the next-hop is unreachable. |
| IBGP routes selected as best routes | Number of "best" routes in the BGP4 route table that are IBGP routes. |
| EBGP routes selected as best routes | Number of "best" routes in the BGP4 route table that are EBGP routes. |

# Examples

The following example shows sample output from the **show ip bgp routes** command.

```
device> show ip bgp routes
Total number of BGP Routes: 97371
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix              Next Hop        MED       LocPrf     Weight Status
1      10.3.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      10.8.1.0/24         192.168.4.106   0           100        0      BE
         AS_PATH: 65001
```

The following example shows sample output from the **show ip bgp routes** command when the **best** keyword is used.

```
device> show ip bgp routes best
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix              Next Hop        MED       LocPrf     Weight Status
1      10.3.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 80
2      10.4.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 1
3      10.60.212.0/22      192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      10.6.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      10.2.0.0/16         192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701
…
```

The following example shows sample output from the **show ip bgp routes** command when the **detail** keyword is used.

```
device> show ip bgp routes detail

Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
1       Prefix: 10.5.5.5/32,  Status: BE,  Age: 0h2m10s
         NEXT_HOP: 10.0.0.1, Metric: 0, Learned from Peer: 10.0.0.1 (3)
          LOCAL_PREF: 100,  MED: none,  ORIGIN: igp,  Weight: 0
         AS_PATH: 3
            Adj_RIB_out count: 2,  Admin distance 20
       Last update to IP routing table: 0h2m10s, 1 path(s) installed:
       Route is advertised to 2 peers:
        10.0.0.3(65002)                        10.0.0.5(65002)
```

The following example shows sample output from the **show ip bgp routes** command when the **summary** keyword is used.

```
device> show ip bgp routes summary

  Total number of BGP routes (NLRIs) Installed     : 20
  Distinct BGP destination networks                : 20
  Filtered BGP routes for soft reconfig            : 100178
  Routes originated by this router                 : 2
  Routes selected as BEST routes                   : 19
  BEST routes not installed in IP forwarding table : 1
  Unreachable routes (no IGP route for NEXTHOP)    : 1
  IBGP routes selected as best routes              : 0
  EBGP routes selected as best routes              : 17
```

The following example shows sample output from the **show ip bgp routes** command when the **unreachable** keyword is used.

```
device> show ip bgp routes unreachable

Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix      Next Hop        Metric     LocPrf     Weight Status
1      10.8.8.0/24  192.168.5.1     0          101        0
         AS_PATH: 65001 4355 1
```

The following example shows sample output from the **show ip bgp routes** command when an IP address is specified.

```
device> show ip bgp route 10.3.4.0

Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH S:SUPPRESSED F:FILTERED
s:STALE
       Prefix       Next Hop       MED   LocPrf  Weight Status
1      10.3.4.0/24  192.168.4.106        100     0      BE
         AS_PATH: 65001 4355 1 1221
       Last update to IP routing table: 0h12m1s, 1 path(s) installed:
         Gateway           Port
         192.168.2.1       1/2/1
       Route is advertised to 1 peers:
        10.20.20.2(65300)
```

# show ip bgp routes community

Displays BGP4 route information that is filtered by community and other options.

## Syntax

**show ip bgp routes community** { *num* | *aa:nn* | **internet** | **local-as** | **no-advertise** | **no-export** }

## Parameters

**community**
Displays routes filtered by a variety of communities.

*num*
Specifies a community number n the range from 1 to 4294967200.

*aa:nn*
Specifies an autonomous system-community number.

**internet**
Displays routes for the Internet community.

**local-as**
Displays routes for a local sub-AS within the confederation.

**no-advertise**
Displays routes with this community that cannot be advertised to any other BGP4 devices at all.

**no-export**
Displays routes for the community of sub-ASs within a confederation.

## Modes

User EXEC mode

# show ip bgp summary

Displays summarized information about the status of all BGP connections.

## Syntax

**show ip bgp summary**

## Modes

User EXEC mode

## Usage Guidelines

If a BGP4 peer is not configured for an address-family, the peer information is not displayed. If a BGP4 peer is configured for an address-family but not negotiated for an address-family after the BGP4 peer is in the established state, the **show ip bgp summary** command output shows (**NoNeg** ) at the end of the line for this peer.

## Command Output

The **show ip bgp summary** command displays the following information:

| This field | Displays |
|---|---|
| Router ID | The device ID. |
| Local AS Number | The BGP4 AS number for the device. |
| Confederation Identifier | The AS number of the confederation in which the device resides. |
| Confederation Peers | The numbers of the local autonomous systems contained in the confederation. This list matches the confederation peer list you configure on the device. |
| Maximum Number of Paths Supported for Load Sharing | The maximum number of route paths across which the device can balance traffic to the same destination. The feature is enabled by default but the default number of paths is 1. You can increase the number from 2 through 8 paths. |
| Number of Neighbors Configured | The number of BGP4 neighbors configured on this device, and currently in established state. |
| Number of Routes Installed | The number of BGP4 routes in the device BGP4 route table and the route or path memory usage. |
| Number of Routes Advertising to All Neighbors | The total of the RtSent and RtToSend columns for all neighbors, the total number of unique ribout group entries, and the amount of memory used by these groups. |
| Number of Attribute Entries Installed | The number of BGP4 route-attribute entries in the device route-attributes table and the amount of memory used by these entries. |
| Neighbor Address | The IP addresses of the BGP4 neighbors for this device. |
| AS# | The AS number. |
| State | The state of device sessions with each neighbor. The states are from this perspective of the device, not the neighbor. State |

| This field | Displays |
|---|---|
| | values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each device:<br>• IDLE - The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process. A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• ADMND - The neighbor has been administratively shut down.<br>• CONNECT - BGP4 is waiting for the connection process for the TCP neighbor session to be completed.<br>• ACTIVE - BGP4 is waiting for a TCP connection from the neighbor. **Note** : If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.<br>• OPEN SENT - BGP4 is waiting for an Open message from the neighbor.<br>• OPEN CONFIRM - BGP4 has received an Open message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the device receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.<br>• ESTABLISHED - BGP4 is ready to exchange UPDATE packets with the neighbor.<br>**Operational States:**<br>Additional information regarding the operational states of BGP described above may be added as described in the following:<br>• **(+)** - is displayed if there is more BGP data in the TCP receiver queue. **Note** : If you display information for the neighbor using the **show ip bgp neighbor***ip-addr* command, the TCP receiver queue value will be greater than 0.<br>• **(>)** - indicates that there is more BGP data in the outgoing queue.<br>• **(-)** - indicates that the session has gone down and the software is clearing or removing routes.<br>• **(*)** - indicates that the inbound or outbound policy is being updated for the peer.<br>• (**c)** - indicates that the table entry is clearing.<br>• (**p)** - indicates that the neighbor ribout group membership change is pending or in progress<br>• **(s)** - indicates that the peer has negotiated restart, and the session is in a stale state.<br>• **(r)** - indicates that the peer is restarting the BGP4 connection, through restart.<br>• **(^)** - on the standby MP indicates that the peer is in the ESTABLISHED state and has received restart capability (in the primary MP).<br>• **(<)** - indicates that the device is waiting to receive the "End of RIB" message the peer. |
| Time | The time that has passed since the state last changed. |
| Accepted | The number of routes received from the neighbor that this device installed in the BGP4 route table. Usually, this number is |

| This field | Displays |
|---|---|
| | lower than the RoutesRcvd number. The difference indicates that this device filtered out some of the routes received in the UPDATE messages. |
| Filtered | The routes or prefixes that have been filtered out:<br>• If soft reconfiguration is enabled, this field shows how many routes were filtered out (not placed in the BGP4 route table) but retained in memory.<br>• If soft reconfiguration is not enabled, this field shows the number of BGP4 routes that have been filtered out. |
| Sent | The number of BGP4 routes the device has sent to the neighbor. |
| ToSend | The number of routes the device has queued to advertise and withdraw to a neighbor. |

# Examples

The following example displays sample output from the **show ip bgp summary** command.

```
device> show ip bgp summary
   BGP4 Summary
  Router ID: 7.7.7.7   Local AS Number: 100
  Confederation Identifier: not configured
  Confederation Peers:
  Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
  Number of Neighbors Configured: 1, UP: 1
  Number of Routes Installed: 0
  Number of Routes Advertising to All Neighbors: 0 (0 entries)
  Number of Attribute Entries Installed: 0
  '+': Data in InQueue '>': Data in OutQueue '-': Clearing
  '*': Update Policy 'c': Group change 'p': Group change Pending
  'r': Restarting 's': Stale '^': Up before Restart '<': EOR waiting
  Neighbor Address  AS#         State   Time          Rt:Accepted Filtered Sent     ToSend
  10.1.1.8          100         ESTAB   0h 9m16s      0           0        0        0
```

# show ip bgp vrf

Displays entries in the IPv4 Border Gateway Protocol (BGP4) routing table for a virtual routing and forwarding (VRF) instance.

## Syntax

**show ip bgp vrf** *vrf-name*

**show ip bgp vrf** *vrf-name ipv6 address/ /mask* [ **longer-prefixes** ]

**show ip bgp vrf** *vrf-name ip address/* [ */mask* ] [ **longer-prefixes** ]

**show ip bgp vrf** *vrf-name* **attribute-entries**

**show ip bgp vrf** *vrf-name* **dampened-paths**

**show ip bgp vrf** *vrf-name* **filtered-routes** [ **detail** ] [ *ip-addr* { **/** *mask* } [ **longer-prefixes** ] ] | **as-path-access-list** *name* ] | **prefix-list** *name* ]

**show ip bgp vrf** *vrf-name* **flap-statistics**

**show ip bgp vrf** *vrf-name* **flap-statistics** *ip-addr* { **/** *mask* } [ **longer-prefix** ]

**show ip bgp vrf** *vrf-name* **flap-statistics as-path-filter** *name*

**show ip bgp vrf** *vrf-name* **flap-statistics neighbor** *ip-addr*

**show ip bgp vrf** *vrf-name* **flap-statistics regular-expression** *name*

**show ip bgp vrf** *vrf-name* **nexthop** [ *ip-addr* | **reachable** | **unreachable** ]

**show ip bgp vrf** *vrf-name* **peer-group** *peer-group-name*

**show ip bgp vrf** *vrf-name* **summary**

## Parameters

*vrf-name*
Specifies the name of a VRF instance.

*ipv6 address /mask*
Specifies an IPv6 address and mask.

**longer-prefixes**
Specifies all statistics for routes that match the specified route, or that have a longer prefix than the specified route.

*ip address /mask*
Specifies an IP address and mask.

**attribute-entries**
Specifies BGP4 route-attribute entries that are stored in device memory.

**dampened-paths**
Specifies multiprotocol BGP (MBGP) paths that have been dampened by route-flap dampening.

**filtered-routes**
Specifies BGP4 filtered routes that are received from a neighbor or peer group.

**detail**
Optionally displays detailed route information.

**as-path-access-list** *name*
Specifies an AS-path ACL. The name must be between 1 and 32 ASCII characters in length.

**prefix-list** *name*
Specifies an IP prefix list. The name must be between 1 and 32 ASCII characters in length.

**flap-statistics**
Specifies the route flap statistics for routes received from or sent to a BGP4 neighbor.

**as-path-filter** *name*
Specifies an AS-path filter.

**neighbor**
Displays flap statistics only for routes learned from the specified neighbor.

*ip-addr*
IPv4 address of the neighbor.

**regular-expression**
Specifies a regular expression in the display output on which to filter.

*name*
Name of an AS-path filter or regular expression.

**nexthop**
Specifies the configured next hop.

**reachable**
Specifies reachable next hops.

**unreachable**
Specifies unreachable next hops.

**peer-group** *peer-group-name*
Specifies a peer group.

**summary**
Displays summarized information.

# Modes

User EXEC mode

# show ip bgp vrf neighbors

Displays configuration information and statistics for BGP4 neighbors of the device for a virtual routing and forwarding (VRF) instance.

## Syntax

**show ip bgp vrf** *vrf-name* **neighbors** [ *ip-addr* ]

**show ip bgp vrf** *vrf-name* **neighbors last-packet-with-error**

**show ip bgp vrf** *vrf-name* **neighbors routes-summary**

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **advertised-routes** [ **detail** ] [ *ip address /mask* ]

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **flap-statistics**

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **last-packet-with-error** [ **decode** ]

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **received** [ **prefix-filter** ]

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **received-routes** [ **detail** ]

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **rib-out-routes** [ **detail** ] [ *ipv6 address /mask* ]

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **routes**

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **routes** { **best** | **not-installed-best** | **unreachable** }

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **routes detail** { **best** | **not-installed-best** | **unreachable** }

**show ip bgp vrf** *vrf-name* **neighbors** *ip-addr* **routes-summary**

## Parameters

*vrf-name*
Specifies the name of a VRF instance.

**neighbors**
Specifies a neighbor.

*ip-addr*
IPv4 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**
Displays the last packet with an error.

**routes-summary**
Displays routes received, routes accepted, number of routes advertised by peer, and so on.

**advertised-routes**
Specifies the routes that the device has advertised to the neighbor during the current BGP4 session.

**detail**
Specifies detailed information.

*ip address /mask*
Specifies an IP address and mask.

**flap-statistics**
Specifies the route flap statistics for routes received from or sent to a BGP4 neighbor.

**last-packet-with-error**
Specifies the last packet with an error.

**decode**
Decodes the last packet that contained an error from any of a device's neighbors.

**received**
Specifies Outbound Route Filters (ORFs) received from BGP4 neighbors of the device.

**prefix-filter**
Displays the results for ORFs that are prefix-based.

**received-routes**
Specifies all route information received in route updates from BGP4 neighbors of the device since the soft-reconfiguration feature was enabled.

**rib-out-routes**
Displays information about the current BGP4 Routing Information Base (Adj-RIB-Out) for specific neighbors and specific destination networks.

**routes**
Displays a variety of route information received in UPDATE messages from BGP4 neighbors.

**best**
Displays routes received from the neighbor that are the best BGP4 routes to their destination.

**not-installed-best**
Displays routes received from the neighbor that are the best BGP4 routes to their destination but were not installed in the route table because the device received better routes from other sources.

**unreachable**
Displays routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.

**routes-summary**
Displays all route information received in UPDATE messages from BGP4 neighbors.

# Modes

User EXEC mode

# show ip bgp vrf routes

Displays statistics for the routes in the BGP4 route table of a device for a virtual routing and forwarding (VRF) instance.

## Syntax

**show ip bgp vrf** *vrf-name* **routes** [ **detail** ] [ *num* | *ip-address/prefix* | **age** *num* | **as-path-access-list** *name* | **as-path-filter** *number* | **best** | **cidr-only** | **community-access-list** *name* | **community-filter** *number* | **community-reg-expression** *expression* | **local** | **neighbor** *ip-addr* | **nexthop** *ip-addr* | **no-best** | **not-installed-best** | **prefix-list** *string* | **regular-expression** *name* | **route-map** *name* | **summary** | **unreachable** ]

## Parameters

*vrf-name*
Specifies the name of a VRF instance.

**detail**
Displays detailed information.

*num*
Table entry at which the display starts. For example, if you want to list entries beginning with table entry 100, specify 100.

*ip-address/prefix*
Specifies an IP address and prefix.

**age** *num*
Displays BGP4 route information that is filtered by age.

**as-path-access-list** *name*
Displays BGP4 route information that is filtered by autonomous system (AS)-path access control list (ACL).

**as-path-filter** *number*
Displays BGP4 route information that is filtered using the specified AS-path filter.

**best**
Displays BGP4 route information that the device selected as best routes.

**cidr-only**
Displays BGP4 routes whose network masks do not match their class network length.

**community-access-list** *name*
Displays BGP4 route information for an AS-path community access list.

**community-filter** *number*
Displays BGP4 route information that matches a specific community filter.

**community-reg-expression** *expression*
Displays BGP4 route information for an ordered community list regular expression.

**local**
Displays BGP4 route information about selected local routes.

**neighbor** *ip-addr*
Displays BGP4 route information about selected BGP neighbors.

**nexthop** *ip-addr*
Displays BGP4 route information about routes that are received from the specified next hop.

**no-best**
Displays BGP4 route information that the device selected as not best routes.

**not-installed-best**
Displays BGP4 route information about best routes that are not installed.

**prefix-list** *string*
Displays BGP4 route information that is filtered by a prefix list.

**regular-expression** *name*
Displays BGP4 route information about routes that are associated with the specified regular expression.

**route-map** *name*
Displays BGP4 route information about routes that use the specified route map.

**summary**
Displays BGP4 summary route information.

**unreachable**
Displays BGP4 route information about routes whose destinations are unreachable through any of the BGP4 paths in the BGP4 route table.

# Modes

User EXEC mode

# show ip bgp vrf routes community

Displays BGP4 route information that is filtered by community and other options for a virtual routing and forwarding (VRF) instance.

## Syntax

**show ip bgp vrf routes community** *vrf-name* { *num* | *aa:nn* | **internet** | **local-as** | **no-advertise** | **no-export** }

## Parameters

**community**
Displays routes filtered by a variety of communities.

*num*
Specifies a community number n the range from 1 to 4294967200.

*aa:nn*
Specifies an autonomous system-community number.

**internet**
Displays routes for the Internet community.

**local-as**
Displays routes for a local sub-AS within the confederation.

**no-advertise**
Displays routes with this community that cannot be advertised to any other BGP4 devices at all.

**no-export**
Displays routes for the community of sub-ASs within a confederation.

## Modes

User EXEC mode

# show ip cache

Displays IP forwarding cache.

## Syntax

**show ip cache** [ **vrf** *vrf-name* ] [ *ip-address* | *index*]

**show ip cache resource**

## Parameters

**vrf** *vrf-name*

Displays cache details for a specific VPN Routing/Forwarding instance.

*ip-address*

Displays cache details for a specific IP address.

*index*

Displays cache details for cache beginning with the row following the number you enter.

**resource**

Displays the number of entries in the cache.

## Modes

User EXEC mode

## Command Output

The **show ip cache** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IP address of the destination. |
| Next Hop | The IP address of the next-hop router to the destination. This field contains either an IP address or the value DIRECT. DIRECT means the destination is either directly attached or the destination is an address on this device. For example, the next hop for loopback addresses and broadcast addresses is shown as DIRECT. |
| MAC | The MAC address of the destination.<br>If the entry is type U (indicating that the destination is this device), the address consists of zeroes. |
| Type | The type of host entry, which can be one or more of the following:<br>• D - Dynamic<br>• P - Permanent<br>• F - Forward<br>• U - Us<br>• C - Complex Filter<br>• W - Wait ARP<br>• I - ICMP Deny<br>• K - Drop |

| Output field | Description |
|---|---|
| | • R - Fragment<br>• S - Snap Encap |
| Port | The port through which this device reaches the destination. For destinations that are located on this device, the port number is shown as "n/a". |
| VLAN | Indicates the VLANs the listed port is in. |
| Pri | The QoS priority of the port or VLAN. |

# Examples

The following example is sample output from the **show ip cache** command.

```
device# show ip cache
Entries in default routing instance:
Total number of cache entries: 1
D:Dynamic  P:Permanent  F:Forward  U:Us  C:Complex Filter
W:Wait ARP  I:ICMP Deny  K:Drop  R:Fragment  S:Snap Encap
     IP Address         Next Hop        MAC             Type    Port    Vlan    Pri
1    192.168.1.11       DIRECT          0000.0000.0000  PU      n/a             0
1    192.168.1.125      DIRECT          0000.0000.0000  PU      n/a             0
1    10.168.1.11        DIRECT          0000.0000.0000  PU      n/a             0
```

The following example is sample output from the **show ip cache resource** command.

```
device# show ip cache resource
9 entries in ip-cache, maximum #: 10000
```

# show ip client-pub-key

Displays the currently loaded public keys.

## Syntax

**show ip client-pub-key**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Examples

The following example displays sample output of the **show ip client-pub-key** command.

```
device(config)# show ip client-pub-key
---- BEGIN SSH2 PUBLIC KEY ----
Comment: DSA Public Key
AAAAB3NzaC1kc3MAAACBAPY8ZOHY2yFSJA6XYC9HRwNHxaehvx5wOJ0rzZdzoSOXxbET W6ToHv8D1UJ/
z+zHo9Fiko5XybZnDIaBDHtblQ+Yp7StxyltHnXF1YLfKD1G4T6JYrdH YI14Om
1eg9e4NnCRleaqoZPF3UGfZia6bXrGTQf3gJq2e7Yisk/gF+1VAAAAFQDb8D5cv
wHWTZDPfX0D2s9Rd7NBvQAAAIEAlN92+Bb7D4KLYk3IwRbXblwXdkPggA4pfdtW9v
GfJ0/RHd+NjB4eo1D+0dix6tXwYGN7PKS5R/FXPNwxHPapcj9uL1Jn2AWQ2dsknf+i/FAA
vioUPkmdMc0zuWoSOEsSNhVDtX3WdvVcGcBq9cetzrtOKWOocJmJ80qadxTRHtUAAACB
AN7CY+KKv1gHpRzFwdQm7HK9bb1LAo2KwaoXnadFgeptNBQeSXG1vO+JsvphVMBJc9HS
n24VYtYtsMu74qXviYjziVucWKjjKEb11juqnF0GDlB3VVmxHLmxnAz643WK42Z7dLM5
sY29ouezv4Xz2PuMch5VGPP+CDqzCM4loWgV
---- END SSH2 PUBLIC KEY ----
```

# show ip dhcp-client options

Displays the list of options the Dynamic Host Configuration Protocol (DHCP) client has received from the DHCP server.

## Syntax

**show ip dhcp-client options**

## Modes

User EXEC mode

Global configuration mode

## Usage Guidelines

The DHCP client must be enabled to be able to view the output of this command correctly.

## Command Output

The **show ip dhcp-client options** command displays the following information:

| Output field | Description |
|---|---|
| DHCP Client Received Option(s) | Specifies the options the client has received, such as the dynamic IP address, subnet mask, lease time, server IP address, default-router address, TFTP server address, boot filename, DNS-server address, host name, and vendor-specific information. |

## Examples

The following example displays the DHCP client options received from the server.

```
device(config)# show ip dhcp-client options
DHCP Client Received Option(s) :

Client Received Options on port: 1/1/2
    Dynamic IP address: 110.1.1.2
           Subnet mask: 255.255.255.0
            Lease Time: 2400
       Server IP Address (option  54): 110.1.1.1
  default-router address (option   3): 110.1.1.1
     TFTP server address (option 150): 110.1.1.20
        TFTP Server Name (option  66): CHN-EL1-E03-65-112
           Boot filename (option  67): boot.cfg
      dns-server address (option   6): 1.1.1.10
         DNS Server Name (option  15): ruckuscom
              Host Name (option  12): dhcp_host
    Vendor Specific Info (option  43): None
```

The following example displays the DHCP client options including the image type and flash location.

```
device(config)# show ip dhcp-client options

DHCP Client Received Option(s) :

Client Received Options on port: 1/1/35
    Dynamic IP address: 65.1.1.2
           Subnet mask: 255.255.255.0
            Lease Time: 86400
       Server IP Address (option  54): 65.1.1.1
  default-router address (option   3): None
     TFTP server address (option 150): 65.1.1.1
        TFTP Server Name (option  66): None
           Boot filename (option  67): FI08070_Manifest.txt router primary
      dns-server address (option   6): None
         DNS Server Name (option  15): None
              Host Name (option  12): None
    Vendor Specific Info (option  43): None
```

The following example displays information for the DHCP client when the vendor specific information (VSI) option information is received from the DHCP server as text (create default ve).

```
device(config)# show ip dhcp-client options

DHCP Client Received Option(s) :

Client Received Options on port: 2/1/21
    Dynamic IP address: 20.1.1.2
           Subnet mask: 255.255.255.0
            Lease Time: 86400
       Server IP Address (option  54): 20.1.1.1
  default-router address (option   3): 20.1.1.1
     TFTP server address (option 150): 20.1.1.1
        TFTP Server Name (option  66): None
           Boot filename (option  67): None
      dns-server address (option   6): 110.1.1.20
         DNS Server Name (option  15): brd.com
    Vendor Specific Info (option  43): create default ve
```

The following example displays information for the DHCP client when the VSI option information is received from the DHCP server in the expected TLV format data (sub-option data in comma-separated IP address format).

```
device(config)# show ip dhcp-client options

DHCP Client Received Option(s) :

Client Received Options on port: 2/1/21
    Dynamic IP address: 20.1.1.2
           Subnet mask: 255.255.255.0
            Lease Time: 86400
       Server IP Address (option  54): 20.1.1.1
  default-router address (option   3): 20.1.1.1
     TFTP server address (option 150): 20.1.1.1
        TFTP Server Name (option  66): None
           Boot filename (option  67): None
      dns-server address (option   6): 110.1.1.20
         DNS Server Name (option  15): brd.com
    Vendor Specific Info (option  43): TLV Format Data
                  (Code 6) : 12.12.12.12 13.13.13.13
```

The following example displays output for the DHCP client when the VSI option information is received from the DHCP server in TLV format, but with sub-option data as non-readable characters.

```
device(config)# show ip dhcp-client options

Client Received Options on port: 2/1/21
    Dynamic IP address: 20.1.1.2
           Subnet mask: 255.255.255.0
            Lease Time: 86400
       Server IP Address (option  54): 20.1.1.1
  default-router address (option   3): 20.1.1.1
     TFTP server address (option 150): 20.1.1.1
        TFTP Server Name (option  66): None
           Boot filename (option  67): None
      dns-server address (option   6): 110.1.1.20
         DNS Server Name (option  15): brd.com
    Vendor Specific Info (option  43): TLV Format Data
                                       ( Code   6 ) :
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.70 | This command was modified to show DHCP client options in the new option format. |
| 08.0.80 | This command was modified to include information in the output for option 67 enhancements where both the image type and flash location are specified by the user. This command was also modified to display information for option 43 enhancements where data related to the VSI received from the SmartZone DHCP server is displayed. |

# show ip dhcp relay information

Displays the configured DHCP relay information options.

## Syntax

**show ip dhcp relay information**

## Modes

Global configuration mode

Privileged EXEC mode

Interface configuration mode

## Usage Guidelines

The outputs of the show commands vary depending on the relay information options you configure. See the examples below for the various outputs.

## Examples

The following example displays the default output if option 82 is not enabled.

```
device(config)# show ip dhcp relay information
Relay Agent Information: format: Circuit-ID: vlan-mod-port
                                 Remote-ID : mac
                                 Policy    : replace
```

The following output displays if only the subscriber ID is configured. The circuit ID and remote ID display the defaults.

```
device(config)# show ip dhcp relay information
Relay Agent Information: policy: replace
 port : 1/2/3:1
     circuit-id      : 000a01020301
     remote-id       : 001094000002
     subscriber-id   : Brcd01
```

The following output displays if all the sub-options are configured. This output is displayed only to the relay connected to the client. The output displays all 63 configured characters for the CID and RID, and 50 characters for the SID.

```
device(config-if-e10000-1/2/3)# show ip dhcp relay information
Relay Agent Information: policy: replace
 port : 1/2/3
     circuit-id      : Brcd01
     remote-id       : remote01
     subscriber-id   : Brcd02
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ip dhcp relay information brief

Displays the configured DHCP relay information options in brief.

## Syntax

**show ip dhcp relay information brief**

## Modes

Global configuration mode

Privileged EXEC mode

Interface configuration mode

## Usage Guidelines

The **show ip dhcp relay information brief** command output shows a maximum of 20 characters. The **show ip dhcp relay information** command displays all the characters.

## Examples

The following output displays when Option 82 is not enabled.

```
device(config)# show ip dhcp relay information brief
Relay Agent Information: policy: replace
```

The following output displays if only the subscriber-id is configured.

```
device(config)# show ip dhcp relay info brief
Relay Agent Information: policy: replace
 Port       Circuit-ID          Remote-ID           Subscriber-ID
 1/2/3      000a01020301        001094000002        Brcd01
```

The following output displays if the circuit ID or remote ID is configured.

```
device(config-if-e10000-1/2/3)# show ip dhcp relay info brief
Relay Agent Information: policy: replace
 Port       Circuit-ID          Remote-ID           Subscriber-ID
 1/2/3          Brcd01             001094000002        none
```

The following example displays if all the sub-options are configured.

```
device(config-if-e10000-1/2/3)# show ip dhcp relay info brief
Relay Agent Information: policy: replace
 Port       Circuit-ID          Remote-ID           Subscriber-ID
 1/2/3          Brcd01             remote01            Brcd02
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ip dhcp-server address-pools

Displays information about Dynamic Host Configuration Protocol (DHCP) address pools.

## Syntax

**show ip dhcp-server address-pools** [*name*]

## Parameters

*name*

Specifies the name of an address pool.

## Modes

Global configuration mode

## Command Output

The **show ip dhcp-server address-pool** command displays the following information:

| Output field | Description |
|---|---|
| Pool name | The name of the address pool |
| Time elapsed since last save | The amount of time that has elapsed since the last save |
| Total number of active leases | The number of leases that are currently active |
| Address Pool State | The state of the address pool (active or inactive) |
| IP Address Exclusions | IP addresses that are not included in the address pool |
| bootfile | The name of the bootfile |
| dhcp-default-router | The address of the DHCP default router |
| dhcp-server-router | The address of the DHCP server router |
| dns-server | The address of the DNS server |
| domain-name | The name of the domain |
| lease | The identifier for the lease |
| ip-telephony-voice-server | The IP address of the voice server |
| ip-telephony-data-server | The IP address of the data server |
| wpad | The network location of the PAC file |
| xwindow manager | The IP addresses of systems that are running the X Window System Display Manager and are available to the client. |
| netbios-name-server | The address of the netBIOS name server |
| network | The address of the network |
| tftp-server | The IP address of the TFTP server |
| next-bootstrap-server | The IP address of the next-bootstrap server |
| vendor-class ascii | The ASCII value of the DHCP client |
| option | The value of the vendor specific information |

# Examples

The following example displays information about all IP DHCP server address pools.

```
device# show ip dhcp-server address-pools
Showing all address pool(s):
Pool Name: one
Time elapsed since last save: 0d:0h:6m:52s
Total number of active leases: 2
Address Pool State: active
IP Address Exclusions: 192.168.1.45
IP Address Exclusions: 192.168.1.99 192.168.1.103
Pool Configured Options:
bootfile: FI08030b_Manifest.txt
dhcp-default-router: 192.168.1.1
dns-server: 192.168.1.100
domain-name: example.com
lease: 0 0 30
ip-telephony-voice-server: MCIPADD=192.168.42.1,MCPORT=1719,TFTPSRVR=192.168.42.1
ip-telephony-data-server: MCIPADD=192.168.42.1,MCPORT=1719,TFTPSRVR=192.168.42.1
wpad: http://172.26.67.243:8080/wpad.dat
xwindow manager: 10.38.12.1 10.38.12.3 10.38.12.5
netbios-name-server: 192.168.1.101
network: 192.168.1.0 255.255.255.0
hostname: ruckus_router
tftp-server:172.26.51.66
next-bootstrap-server: 192.168.1.102
vendor-class ascii:  "Ruckus CPE"
option: 43 hex 0108c0a80a01c0a81401
```

The following example displays information about all IP DHCP server address pools including the image type and flash location that have been specified by the user.

```
device# show ip dhcp-server address-pools

Showing all address pool(s):

                    Pool Name:  temp
 Time elapsed since last save:  00d:00h:07m:08s
Total number of active leases:  0
           Address Pool State:  active

      Pool Configured Options:
                          lease:  1 0 0
                        network:  65.1.1.0 255.255.255.0
  option  67 (Bootfile-Name      ):  ascii "fi8070_manifest.txt router primary"
 option 150 (TFTP Server Addr   ):  ip 50.50.50.1
```

# History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was modified to include X Window System Display Manager updates in the output. |
| 08.0.30mb | This command was modified to include the vendor class option in the output. |
| 08.0.40 | This command was modified to include updates in the output for WPAD, IP-telephony-voice, and data server. |
| 08.0.80 | This command was modified to include information in the output for option 67 enhancements where both the image type and flash location are specified by the user. |

# show ip dhcp-server binding

Displays the IP DHCP server lease entry.

## Syntax

**show ip dhcp-server binding**

## Modes

Global configuration mode.

## Usage Guidelines

The **show ip dhcp-server binding** command displays a specific DHCP active lease, or all active leases.

## Command Output

The **show ip dhcp-server binding** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IP addresses currently in the binding database. |
| Client ID/Hardware address | The hardware address of the client. |
| Lease expiration | The time when this lease will expire. |
| Type | The type of lease. |

## Examples

The following example displays the IP DHCP server bindings.

```
device# show ip dhcp-server binding
Bindings from all pools:
IP Address Client-ID/ Lease expiration Type
Hardware address
192.168.1.2 0000.005d.a440 0d:0h:29m:31s Automatic
192.168.1.3 0000.00e1.26c0 0d:0h:29m:38s Automatic
```

# show ip dhcp-server flash

Displays the lease-binding database stored in the flash memory.

## Syntax

**show ip dhcp-server flash**

## Modes

Global configuration mode

User EXEC mode

## Command Output

The **show ip dhcp-server flash** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IP address of the flash memory lease-binding database. |
| Client-ID/Hardware address | The address of the client. |
| Lease expiration | The time when the lease will expire. |
| Type | The type of lease. |

## Examples

The following example displays details of the lease-binding database stored in the flash memory.

```
device# show ip dhcp-server flash
Address Pool Binding:
IP Address Client-ID/ Lease expiration Type
Hardware address
192.168.1.2 0000.005d.a440 0d:0h:18m:59s Automatic
192.168.1.3 0000.00e1.26c0 0d:0h:19m:8s Automatic
```

# show ip dhcp-server statistics

Displays DHCP server statistics for a specific pool or all pools.

## Syntax

**show ip dhcp-server statistics** [ *pool-name* ]

## Parameters

*pool-name*

Specifies a pool in ASCII characters.

## Modes

Privileged EXEC mode.

## Usage Guidelines

The **show ip dhcp-server summary** command displays packet counters that are received to the DHCP server for a specified pool or all pools. DHCP must be enabled before this command can be executed.

## Examples

The following example displays DHCP server statistics for a specified pool.

```
device# show ip dhcp-server statistics test

Statistics for address pool:
DHCP server pool name :  test(active)
Dynamically allocated addresses :  1
Statically allocated addresses :   1
      Total allocated addresses :   2
  Received Packets
                           (Valid)  (Dropped)
              DHCP-DISCOVER :  1     2
               DHCP-REQUEST :  1     0
               DHCP-DECLINE :  0     0
               DHCP-RELEASE :  1     0
                DHCP-INFORM :  0     0
    Total Packets Received  :  4
  Sent Packets
                 DHCP-OFFER :  1
                   DHCP-ACK :  2
                   DHCP-NAK :  0
Total Packets Transmitted  :  3
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show ip dhcp-server summary

Displays the IP DHCP server summary.

## Syntax

**show ip dhcp-server summary**

## Modes

Global configuration mode.

User EXEC mode.

## Usage Guidelines

The **show ip dhcp-server summary** command displays information about active leases, deployed address pools, undeployed address pools, and server uptime.

## Command Output

The **show ip dhcp-server summary** command displays the following information:

| Output field | Description |
|---|---|
| Total number of active leases | Indicates the number of leases that are currently active. |
| Total number of deployed address-pools | The number of address pools currently in use. |
| Total number of undeployed address-pools | The number of address pools being held in reserve. |
| Server uptime | The amount of time that the server has been active. |

## Examples

The following example displays the IP DHCP server summary.

```
device# show ip dhcp-server summary
DHCP Server Summary:
Total number of active leases: 2
Total number of deployed address-pools: 1
Total number of undeployed address-pools: 0
Server uptime: 0d:0h:8m:27s
```

# show ip dhcp snooping flash

Displays the DHCP snooping learned entries from the flash file.

## Syntax

**show ip dhcp snooping flash**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show ip dhcp snooping flash** command displays the following information:

| Output field | Description |
|---|---|
| DHCP snooping Info | Displays information about the saved DHCP entries in the flash file. This includes details about the total number of learned entries along with the IP address, MAC address, port number, VLAN, lease, and VRF name of each entry. |

## Examples

The following example displays the IP DHCP snooping flash information.

```
device# show ip dhcp snooping flash
Dhcp snooping Info
Total learnt entries 10
SAVED DHCP ENTRIES IN FLASH
   IP Address  Mac Address    Port          Virtual Port vlan  lease VRF
1  10.1.1.20   0000.0000.0001 1/1/1*2/1/25  v100         100   170   default-vrf
2  10.1.1.21   0000.0000.0002 1/1/1*2/1/25  v100         100   170   default-vrf
3  10.1.1.22   0000.0000.0003 1/1/1*2/1/25  v100         100   170   default-vrf
4  10.1.1.23   0000.0000.0004 1/1/1*2/1/25  v100         100   170   default-vrf
5  10.1.1.24   0000.0000.0005 1/1/1*2/1/25  v100         100   170   default-vrf
6  10.1.1.25   0000.0000.0006 1/1/1*2/1/25  v100         100   170   default-vrf
7  10.1.1.26   0000.0000.0007 1/1/1*2/1/25  v100         100   170   default-vrf
8  10.1.1.27   0000.0000.0008 1/1/1*2/1/25  v100         100   170   default-vrf
9  10.1.1.28   0000.0000.0009 1/1/1*2/1/25  v100         100   170   default-vrf
10 10.1.1.29   0000.0000.000a 1/1/1*2/1/25  v100         100   170   default-vrf
```

## History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was introduced. |

# show ip dhcp snooping info

Displays the DHCP snooping binding database.

## Syntax

**show ip dhcp snooping info**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

Beginning with FastIron release 08.0.30b, this command reads data from the DHCP binding database, and not from the flash file, as in releases prior to 08.0.30b.

## Examples

The following example displays the DHCP snooping information.

```
device# show ip dhcp snooping info
Dhcp snooping Info

Total learnt entries 64

Learnt DHCP Snoop Entries

IP Address      Mac Address    Port      Virtual Port vlan  lease   VRF

3.1.1.5         0000.7e49.6183 lg256     v3           3     585    default-vrf
```

## History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was modified to include the output on a switch image. |
| 08.0.61 | The command output was modified. |

# show ip dhcp snooping vlan

Displays the DHCP snooping status for a VLAN and the trusted or untrusted ports.

## Syntax

**show ip dhcp snooping vlan** *vlan-id*

## Parameters

*vlan-id*

Specifies the VLAN ID.

## Modes

Privileged EXEC mode

## Command Output

The **show ip dhcp snooping vlan** command displays the following information:

| Output field | Description |
|---|---|
| IP DHCP snooping VLAN # | Displays whether the IP DHCP snooping is enabled or disabled. |

## Examples

The following example displays the IP DHCP snooping status on VLAN 2.

```
device# show ip dhcp snooping vlan 2
IP DHCP snooping VLAN 2: Enabled
```

# show ip igmp group

Displays the status of IGMP multicast groups on a device.

## Syntax

**show ip igmp** [**vrf** *vrf-name* ] **group** [ *group-address* [ **detail** | **tracking** ] ]

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

*group-address*

Specifies the address of the specific multicast group. If you do not specify a group address, information for all multicast groups is displayed.

**detail**

Displays information for the source list of the multicast group.

**tracking**

Displays information about interfaces that have tracking enabled.

## Modes

Privileged EXEC mode

## Command Output

The **show ip igmp group** command displays the following information:

| Output Field | Description |
|---|---|
| Group | The address of the multicast group |
| Port | The physical port on which the multicast group was received. |
| Intf | The virtual interface on which the multicast group was received. |
| Timer | Shows the number of seconds the interface can remain in exclude mode. An exclude mode changes to include mode if it does not receive an “IS_EX” or “TO_EX” message during a certain period of time. The default is 140 seconds. |
| Mode | Indicates current mode of the interface: include or exclude. If the interface is in Include mode, it admits traffic only from the source list. If an interface is in exclude mode, it denies traffic from the source list and accepts the rest. |
| Srcs | Identifies the source list that will be included or excluded on the interface.<br><br>If IGMP V2 group is in exclude mode with a #_src of 0, the group excludes traffic from 0 (zero) source list, which means that all traffic sources are included. |

# Examples

The example displays information for all IGMP multicast groups.

```
device# show ip igmp group
Total 2 entries
------------------------------------------------------------
Idx Group Address    Port       Intf   Mode      Timer  Srcs
---+---------------+------+------+-------+-----+-----+----
  1 232.0.0.1        e1/1/1   v30     include     0    7
  2 226.0.0.1        e1/1/2   v30     exclude    240   2
                     e1/1/3   e1/1/3  include     0    3
Total number of groups 2
```

The following example displays information for the IGMP multicast group, 239.0.0.1.

```
device# show ip igmp group 239.0.0.1 detail
Total 2 entries
-----------------------------------------------------
Idx Group Address    Port    Intf   Mode    Timer  Srcs
---+---------------+------+------+-------+-----+------
  1 226.0.0.1        e1/1/2   v30    exclude   218   2
     S: 40.40.40.12
     S: 40.40.40.11
     S: 40.40.40.10
     S: 40.40.40.2          (Age: 218)
     S: 40.40.40.3          (Age: 218)
    226.0.0.1        e1/1/3  e1/1/3   include     0   3
     S: 30.30.30.3          (Age: 165)
     S: 30.30.30.2          (Age: 165)
     S: 30.30.30.1          (Age: 165)
```

The following example displays the list of clients that belong to a particular IGMP multicast group.

```
device# show ip igmp group 224.1.10.1 tracking
Total 2 entries
-----------------------------------------------------------
Idx Group Address    Port    Intf    Mode    Timer  Srcs
---+---------------+------+------+-------+-------+-----
  1 226.0.0.1        e1/1/1   v30    exclude   253   3
     S: 40.40.40.12
     S: 40.40.40.11
     S: 40.40.40.10
     S: 40.40.40.2          (Age: 253)
                    C: 10.10.10.1       (Age: 253)
    S: 40.40.40.3          (Age: 253)
                    C: 10.10.10.1       (Age: 253)
    226.0.0.1        e1/1/3   e1/1/3   include     0   3
     S: 30.30.30.3          (Age: 196)
                    C: 10.2.0.1         (Age: 196)
     S: 30.30.30.2          (Age: 196)
                    C: 10.2.0.1         (Age: 196)
     S: 30.30.30.1          (Age: 196)
                    C: 10.2.0.1         (Age: 196)
```

# show ip igmp interface

Displays the status of a multicast enabled port.

## Syntax

**show ip igmp** [ **vrf** *vrf-name* ] **interface** [ **ve** *ve-num* [ **group** *A.B.C.D* ] | **ethernet** *unit**/**slot**/**port* | **tunnel** *tunnel-id* ]

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

**ve** *ve-num*

Specifies displaying information for a specific virtual routing interface.

**group** *A.B.C.D*

Specifies displaying information for a specific group address.

**ethernet** *unit**/**slot**/**port*

Specifies displaying information for an Ethernet interface.

**tunnel** *tunnel-id*

Specifies displaying information about a GRE tunnel interface that is being configured. The GRE tunnel interface is enabled under the router PIM configuration.

## Modes

Privileged EXEC mode

## Command Output

The **show ip igmp interface** command displays the following information:

| Output Field | Description |
|---|---|
| Intf | The virtual interface on which IGMP is enabled. |
| Port | The physical port on which IGMP is enabled. |
| Groups | The number of groups that this interface or port has membership. |
| Version | |
| Oper | The IGMP version that is operating on the interface. |
| Cfg | The IGMP version that is configured for this interface. |
| Querier | Where the Querier resides:<br>The IP address of the router where the querier is located<br>or<br>Self - if the querier is on the same router as the intf or port. |
| Max response | |
| oQrr | Other Querier present timer. |

| Output Field | Description |
|---|---|
| GenQ | General Query timer |
| V1Rtr | Whether IGMPv1 is present on the intf or port. |
| V2Rtr | Whether IGMPv2 is present on the intf or port. |
| Tracking | Fast tracking status:<br>Enabled<br>or<br>Disabled |

# Examples

The following example displays information for a multicast enabled port.

```
device# show ip igmp interface
---------+------+---------+---------------+---------+-----+-----+---------
Intf/Port|Groups| Version |Querier        |  Timer  |V1Rtr|V2Rtr|Tracking
         |      |Oper  Cfg|               |OQrr GenQ|     |     |
---------+------+----+----+---------------+----+----+-----+-----+---------
e1/1/3       1    3    3      Self            0   94   No     No   Disabled
e1/1/4       0    2    -      Self            0   94   No     No   Disabled
v30          1    3    3                                          Disabled
e1/1/2       3    -    -      Self            0   20   No     No
v40          0    3    3                                          Disabled
e1/1/2       3    -    -      Self            0   20   No     No
v50          0    2    -                                          Disabled
e1/1/12      2    -    -      Self            0   29   No     No
e1/1/8       2    -    -      50.1.1.10       46   0   No     Yes
e1/1/1       2    -    -      Self            0   115  No     Yes
```

The following example displays information for the interface VE 4041 group.

```
device# show ip igmp interface ve 4041 group
Total 100 groups
------------------------------------------------------------------------
Idx  Group Address    Port      Intf      GrpCmpV  Mode    Timer  Srcs
----+---------------+---------+---------+-------+-------+-----+-----
 1    239.0.1.1        e1/2/8    v4041     Ver2    exclude  247    0

Total number of groups 1
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to include the IGMP **group** keyword. |

# show ip igmp proxy

Displays information about the proxy groups and interfaces on the default VRF or, when the **vrf** keyword is specified, other VRFs.

## Syntax

**show ip igmp** [ **vrf** *vrf-name* ] **proxy** [ **group** *group-addr* ]

**show ip igmp** [ **vrf** *vrf-name* ] **proxy** [ **interface** { **ethernet** *stack**/**slot**/**port* | **tunnel** *tunnel-id* | **ve** *ve-num* } [ **detail** | **group-resp** *ip-addr* | **stats** ] ]

**show ip igmp** [ **vrf** *vrf-name* ] **proxy** [ **resource** ]

**show ip igmp** [ **vrf** *vrf-name* ] **proxy** [ **stats** ]

**show ip igmp** [ **vrf** *vrf-name* ] **proxy** [ **summary** ]

## Parameters

**vrf** *vrf-name*
Displays information for a VRF instance.

**proxy**
Displays information about the proxy groups and interfaces.

**group** *group-addr*
Displays information for the specified IGMP group.

**interface**
Displays information for the specified interface.

**ethernet** *stack**/**slot**/**port*
Displays information for the specified Ethernet interface.

**tunnel** *tunnel-id*
Displays information for the specified tunnel interface.

**ve** *ve-num*
Displays information for the specified VE interface.

**detail**
Displays detailed information.

**group-resp** *ip-addr*
Displays information for the group response tree.

**stats**
Displays information on the interface status.

**resource**
Displays memory status of various pools.

**summary**
Displays summary information.

**stats**

Displays nformation about queries and reports on a specific interface.

# Modes

Privileged EXEC mode

# Command Output

The **show ip igmp proxy** command displays the following information:

| Output Field | Description |
|---|---|
| Address | Group address. |
| Mode | Multicast group mode. Can be “exclude” or “include.” |
| Source count | Number sources in the given mode. A group in IGMP v2 has exclude mode with zero sources. |
| ref count | Number of proxy interfaces where the responses (query, state, change, etc) are scheduled. |
| flags | Can be “0” or “1”. “1” indicates that the group state has changed and it needs to be reevaluated before a response is generated. “0” indicates that no change in state response is scheduled. |
| Name | Interface name. |
| Oper version | Current querier version or configured version. |
| Cfg Robust | Configured robustness value. |
| Unsoli Interval | Unsolitcited report interval in seconds. |
| Filter Acl Id | Number of the access list. |
| Filter Name | Name of the access list. |

The **show ip igmp proxy summary** command displays the following information:

| Output Field | Description |
|---|---|
| Inst-Name | Number of the proxy instance. |
| Total Grps | NUmber of proxy groups. |

The **show ip igmp proxy stats** command displays the following information:

| Output Field | Description |
|---|---|
| Intf | Interface |
| genQv1 RX | IGMP v1 general query received on proxy interface. |
| genQv2 RX | IGMP v2 general query received on proxy interface. |
| genQv3 RX | IGMP v3 general query received on proxy interface. |
| GrpQ RX | Group query received. |
| SrcQ RX | Source query received. |
| Rprtv1 TX | IGMP v1 report generated. |
| Rprtv2 TX | IGMP 2 report generated. |
| Rprtv3 TX | IGMP v3 report generated. |

| Output Field | Description |
|---|---|
| leave TX | IGMP v2 leave generated. |

# Examples

The following example shows information about the proxy groups and interfaces on the default VRF.

```
device# show ip igmp proxy
Proxy instance name: default-vrf
Total proxy groups: 4
Address             Mode      Source    ref       flags
                              count     count
-------------------------------------------------
225.1.1.1           exclude   0         0         0
225.1.1.2           exclude   0         0         0
225.1.1.3           exclude   0         0         0
225.1.1.4           exclude   0         0         0
Proxy interfaces
----------------
Name      Oper     Cfg       Unsoli    Filter    Filter
          Version  Robust    Interval  Acl Id    Name
-------------------------------------------------
e1/1/3      2         2         1         0
```

The following example shows summary information about the proxy groups and interfaces on the default VRF.

```
device# show ip igmp proxy summary
Proxy instances:
----------------------------------
Inst-Name      Total Grps
----------------------------------
default-vrf    4
```

This example shows information about queries and reports on interface v300.

```
device# show ip igmp proxy stats
Intf      genQv1  genQv2  genQv3  GrpQ    SrcQ    Rprtv1  Rprtv2  Rprtv3  leave
           RX      RX      RX      RX      RX      TX      TX      TX      TX
-------------------------------------------------------------------------------
v3000      0       0       0       0       0       0       0       0       0
```

# show ip igmp settings

Displays global IGMP settings or IGMP settings for a specified VRF.

## Syntax

**show ip igmp** [**vrf** *vrf-name* ] **settings**

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ip igmp settings** command displays the following information:

| Output Field | Description |
|---|---|
| Query Interval | How often the router will query an interface for group membership. |
| Configured Query Interval | The query interval that has been configured for the router. |
| Max Response Time | The length of time in seconds that the router will wait for an IGMP (V1 or V2) response from an interface before concluding that the group member on that interface is down and removing it from the group. |
| Group Membership Time | The length of time in seconds that a group will remain active on an interface in the absence of a group report. |
| Configured Version | The IGMP version configured on the router. |
| Operating Version | The IGMP version operating on the router. |
| Robustness Variable | The Robustness Variable allows tuning for the expected packet loss on a network. If a network is expected to be lossy, the Robustness Variable may be increased. IGMP is robust to (Robustness Variable -1) packet losses. The Robustness Variable must not be zero, and should not be one. Default: 2 |
| Router Alert Check | IGMP (v2/v3) messages have a router-alert option in the IP header. By default this is validated by the router and it drops the packets without the router-alert option. If this check is disabled, IGMP messages without the router-alert option are accepted. |
| Last Member Query Interval | The Last Member Query Interval is the Max Response Time used to calculate the Max Resp Code inserted into Group-Specific Queries sent in response to Leave Group messages. It is also the Max Response Time used in calculating the Max Resp Code for Group-and-Source-Specific Query messages. Default: 10 (1 second) |
| Last Member Query Count | The Last Member Query Count is the number of Group-Specific Queries sent before the router assumes there are no local |

| Output Field | Description |
|---|---|
| | members. The Last Member Query Count is also the number of Group-and-Source-Specific Queries sent before the router assumes there are no listeners for a particular source. Default: the Robustness Variable. |
| Older Host Present Timer | The Older Host Present Interval is the time-out for transitioning a group back to IGMPv3 mode when an older version report is sent for that group. When an older version report is received, routers set their Older Host Present Timer to Older Host Present Interval.<br><br>This value must be ((the Robustness Variable) times (the Query Interval)) plus (one Query Response Interval). |
| Maximum Group Address | This value indicates the maximum number of group address that can be accepted by the router. |

# Examples

The following example shows global IGMP settings.

```
device# show ip igmp settings
IGMP Global Configuration
  Query Interval            : 125s   Configured Interval    : 125
  Max Response Time         : 10s
  Group Membership Time     : 260s
  Operating Version         : 2      Configured Version     : 0
  Robustness Variable       : 2
  Router Alert Check        : Enabled
  Last Member Query Interval: 1      Last Member Query Count: 2
  Older Host Present Timer  : 260
  Maximum Group Address     : 4096
```

# show ip igmp ssm-map

Displays the association between a configured access control list (ACL) and source address mapped to it.

## Syntax

**show ip igmp** [**vrf** *vrf-name* ] **ssm-map** [ *group-address* ]

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

*group-address*

Specifies displaying the ACL ID that has the specified multicast group address in its permit list and listing the source addresses mapped to the specified multicast group address.

## Modes

Privileged EXEC mode

## Examples

The following example shows the association between a configured ACL and source address mapped to it.

```
device# show ip igmp ssm-map
+---------+-----------------+
| Acl id  | Source Address  |
+---------+-----------------+
        20          1.1.1.1
       100          1.1.1.1
        20          2.2.2.2
        20          2.2.2.3
        20          2.2.2.4
        20          2.2.2.5
        20          2.2.2.6
```

The following example shows the ACL IDs that have the specified multicast group address in their permit lists and lists the source addresses mapped to them.

```
device# show ip igmp ssm-map
+---------+-----------------+
| Acl id  | Source Address  |
+---------+-----------------+
        20          1.1.1.1
       100          1.1.1.1
        20          2.2.2.2
        20          2.2.2.3
        20          2.2.2.4
        20          2.2.2.5
        20          2.2.2.6
```

The following example shows the ACL IDs that have the specified multicast group address in their permit lists and lists the source addresses mapped to it.

```
device# show ip igmp ssm-map 232.1.1.1
+---------+-----------------+
| Acl id  | Source Address  |
+---------+-----------------+
        20          1.1.1.1
       100          1.1.1.1
        20          2.2.2.2
        20          2.2.2.3
        20          2.2.2.4
        20          2.2.2.5
        20          2.2.2.6
```

# show ip igmp static

Displays information about static IGMP groups.

## Syntax

**show ip igmp** [ **vrf** *vrf-name* ] **static**

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ip igmp static** command displays the following information:

| Output Field | Description |
|---|---|
| Group Address | The address of the multicast group. |
| Interface Port List | The physical ports on which the multicast groups are received. |

## Examples

The following example shows information about static IGMP groups for the VRF named my_vrf.

```
device#show ip igmp vrf my_vrf static
Group Address    Interface Port List
---------------+---------+---------
      229.1.0.12     1/1/1 ethe 1/1/1
      229.1.0.13     1/1/1 ethe 1/1/1
      229.1.0.14     1/1/1 ethe 1/1/1
      229.1.0.92     1/1/1 ethe 1/1/1
```

# show ip igmp traffic

Displays the traffic status on each virtual routing interface.

## Syntax

**show ip igmp** [**vrf** *vrf-name* ] **traffic**

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ip igmp traffic** command displays the following information:

| Output Field | Description |
|---|---|
| QryV2 | Number of general IGMP V2 query received or sent by the virtual routing interface. |
| QryV3 | Number of general IGMP V3 query received or sent by the virtual routing interface. |
| G-Qry | Number of group specific query received or sent by the virtual routing interface. |
| GSQry | Number of source specific query received or sent by the virtual routing interface. |
| MbrV2 | The IGMP V2 membership report. |
| MbrV3 | The IGMP V3 membership report. |
| Leave | Number of IGMP V2 "leave" messages on the interface. (See ToEx for IGMP V3.) |
| IsIN | Number of source addresses that were included in the traffic. |
| IsEX | Number of source addresses that were excluded in the traffic. |
| ToIN | Number of times the interface mode changed from exclude to include. |
| ToEX | Number of times the interface mode changed from include to exclude. |
| ALLOW | Number of times that additional source addresses were allowed or denied on the interface: |
| BLK | Number of times that sources were removed from an interface. |

# Examples

The following example shows the traffic status on each virtual routing interface.

```
device# show ip igmp traffic
Recv  QryV2 QryV3 G-Qry GSQry MbrV2 MbrV3 Leave  IsIN  IsEX ToIN ToEX ALLOW BLK
v5       29     0     0     0     0     0     0     0     0    0    0     0    0
v18      15     0     0     0     0    30     0    60     0    0    0     0    0
v110      0     0     0     0     0    97     0   142    37    2    2     3    2
Send  QryV1 QryV2 QryV3 G-Qry GSQry
v5        0     2     0     0     0
v18       0     0    30    30     0
v110      0     0    30    44    11
```

# show ip interface

Displays useful information about the configuration and status of the IP protocol and its services, on all interfaces.

## Syntax

**show ip interface** [ **ethernet** *unit*/*slot*/*port* | **loopback** *num* | **tunnel** *num* | **ve** *num* ]

## Parameters

**ethernet** *unit slot port*

Displays the specified Ethernet interface by unit, slot, and port number.

**loopback** *num*

Displays the loopback interface number.

**tunnel** *num*

Displays the tunnel interface number.

**ve** *num*

Displays the Virtual Ethernet interface number.

## Modes

User EXEC mode

## Command Output

The **show ip interface** command displays the following information:

| Output field | Description |
|---|---|
| Interface | The type and the slot and port number of the interface. |
| IP-Address | The IP address of the interface. |
| OK? | Whether the IP address is configured on the interface. |
| Method | Whether the IP address is saved in NVRAM. If you have set the IP address for the interface in the CLI, the Method field is "manual". |
| Status | The link status of the interface. If the user has disabled the interface with the **disable** command, the entry in the 'Status' field is "administratively DOWN". Otherwise, the entry in the 'Status' field is either UP or DOWN. |
| Protocol | Whether the interface can provide two-way communication. If the IP address is configured and the link status of the interface is up, the entry in the 'Protocol' field is UP. Otherwise, the entry in the 'Protocol' field is DOWN. |
| VRF | Whether the VRF is configured or set to default. |

# Examples

The following example displays information about all IP interfaces.

```
device# show ip interface

Interface     IP-Address      OK?  Method   Status    Protocol   VRF
Eth 1/1/6     10.53.5.1       YES  manual   down      down       default-vrf
Eth mgmt1     10.25.224.194   YES  manual   up        up         default-vrf
```

The following example displays the **show ip interface** command specifically for tunnel interface 64.

```
device# show ip interface tunnel 64

Interface Tunnel 64
 port enabled
 port state: UP
 ip address: 10.224.64.0/31
 Port belongs to VRF: default-vrf
 encapsulation: GRE, mtu: 1476, metric: 1
 directed-broadcast-forwarding: disabled
 proxy-arp: disabled
 ip arp-age: 10 minutes
 No Helper Addresses are configured.
 No inbound ip access-list is set
 No outgoing ip access-list is set
```

The following example displays the IP interface VE configurations.

```
device(config)# show ip interface ve 10

Interface Ve 10
    members: ethe 1/1/47 to 1/1/48 ethe 3/1/47
    active: ethe 1/1/47 to 1/1/48 ethe 3/1/47
    port enabled
    port state: UP
    ip address: 100.1.1.1         subnet mask: 255.255.255.0
    Port belongs to VRF: default-vrf
    encapsulation: ETHERNET, mtu: 1500, metric: 1
    directed-broadcast-forwarding: disabled
    ICMP redirect: enabled
    proxy-arp: disabled
    ip arp-age:  10 minutes
    delay notification timer: 20 seconds
    No Helper Addresses are configured.
    No inbound ip access-list is set
    No outgoing ip access-list is set
```

The following example displays the **show ip interface** command to verify a user-configured MAC address. The “ip-mac:” text is followed by the configured MAC address.

```
device# show ip interface ethernet 1/1/6

Interface Ethernet 1/1/6
  port enabled
  port state: DOWN
  ip address: 10.53.5.1         subnet mask: 255.255.255.0
  Port belongs to VRF: default-vrf
  encapsulation: ETHERNET, mtu: 1500, metric: 1
  directed-broadcast-forwarding: disabled
  ICMP redirect: disabled
  proxy-arp: disabled
  ip arp-age:  10 minutes
  No Helper Addresses are configured.
  No inbound ip access-list is set
  No outgoing ip access-list is set
  ip-mac: aaaa.bbbb.cccc
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | The command output was modified to display a user-configured MAC address for an IP interface. |

# show ip mroute

Displays information on multicast routes. You can specify whether you want to display information from static or connected mroutes or from a particular mroute.

## Syntax

**show ip mroute** [**vrf** *vrf-name* ] { **static** | **connected** | **nexthop** | *ip-subnet* [ *mask*]}

## Parameters

**vrf** *vrf-name*
Specifies a VRF route.

**static**
Specifies a static multicast route.

**connected**
Specifies a directly attached (connected) multicast route.

**nexthop**
Specifies an IPv4 next hop table.

*ip-subnet* [ *mask* ]
Specifies an IP address.

## Modes

Privileged EXEC mode

Global configuration mode

## Examples

The following example displays information for IP multicast routes:

```
Device(config)# show ip mroute

Total number of IP routes: 5
 Type Codes - B:BGP  D:Connected  S:Static;   Cost - Dist/Metric
        Destination             Gateway             Port            Cost          Type    Uptime
1       20.20.20.0/24           220.220.220.1       ve 220          1/1            S      8m54s
2       50.50.50.0/24           DIRECT              ve 50           0/0            D      8h26m
3       77.1.1.1/32             DIRECT              loopback 1      0/0            D      8h26m
4       129.129.129.0/24        DIRECT              ve 129          0/0            D      8h26m
5       220.220.220.0/24        DIRECT              ve 220          0/0            D      2h49m
```

The following example displays information for static multicast routes:

```
Device(config)# show ip mroute static

 Type Codes - B:BGP  D:Connected  S:Static;   Cost - Dist/Metric
        Destination             Gateway           Port           Cost    Type    Uptime
1       20.20.20.0/24           220.220.220.1     ve 220         1/1     S       8m54s
```

The following example displays information for directly attached multicast routes:

```
Device(config)# show ip mroute connected

 Type Codes - B:BGP  D:Connected  S:Static;   Cost - Dist/Metric
        Destination             Gateway           Port          Cost    Type    Uptime
1       50.50.50.0/24           DIRECT            ve 50         0/0     D       8h26m
2       77.1.1.1/32             DIRECT            loopback 1    0/0     D       8h26m
3       129.129.129.0/24        DIRECT            ve 129        0/0     D       8h26m
4       220.220.220.0/24        DIRECT            ve 220        0/0     D       2h49m
```

The following example displays information for IP multicast route 50.50.50.100:

```
Device(config)# show ip mroute 50.50.50.100

 Type Codes - B:BGP  D:Connected  S:Static;   Cost - Dist/Metric
        Destination             Gateway           Port          Cost    Type    Uptime
1       50.50.50.0/24           DIRECT            ve 50         0/0     D       8h26m
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# show ip msdp mesh-group

Displays the details of a specific mesh-group.

## Syntax

**show ip msdp** [ **vrf** *vrf-name* ] **mesh-group** *group-name*

## Parameters

**vrf**
Displays the mesh-group details for the VRF instance specified by the *vrf-name* variable.

*vrf-name*
Specifies the VRF instance.

**mesh-group**
Specifies the MSDP group.

*group-name*
Specifies the mesh group.

## Modes

Privileged EXEC mode

Global configuration mode

MSDP router configuration mode

## Usage Guidelines

If used without specifying a VRF, this command shows data from the default VRF.

## Command Output

The **show ip msdp** [ **vrf** *vrf-name* ] **mesh-group** *group-name* command displays the following information:

| Output field | Description |
|---|---|
| Peer Address | The IP address of the MSDP peer that is placed in the mesh group. |
| State | The state of the MSDP device connection with the mesh group. The state can be one of the following:<br>• CONNECT - The session is in the active open state.<br>• ESTABLISH - The MSDP session is fully up.<br>• IDLE - The session is idle.<br>• LISTEN - The session is in the passive open state. |
| KA (Keep Alive) In | The number of MSDP keepalive messages received by the mesh group. |
| KA (Keep Alive) Out | The number of MSDP keepalive messages sent by the mesh group. |
| SA (Source-Active) In | The number of SA messages received by the mesh group. |

| Output field | Description |
|---|---|
| SA (Source-Active) Out | The number of SA messages sent by the mesh group. |
| NOT (Notification) In | The number of notification messages received by the mesh group. |
| NOT (Notification) out | The number of notification messages sent by the mesh group. |
| Age | The number of seconds the messages has been in the cache. |

# Examples

The following example shows the mesh-group configuration details.

```
device#show ip msdp mesh-group
Mesh-Group-Name                      Peer-IP-Address
group1                                 40.0.0.40
group2                                 21.0.0.23
```

The following example shows the details of mesh-group group1.

```
device#show ip msdp mesh-group group1
MSDP MESH-GROUP:group1
 KA: Keepalive SA:Source-Active NOT: Notification
 Peer Address    State          KA              SA             NOT          Age
                                In     Out     In     Out     In     Out
 40.0.0.40       ESTABLISH    1407    1406      0      0      0      0       6
```

The following example shows the mesh-group configuration details for the VRF 10 instance.

```
device#show ip msdp vrf 10 mesh-group
Mesh-Group-Name                      Peer-IP-Address
group1                                 22.0.0.22
group2                                 21.0.0.23
```

The following example shows the mesh-group group2 details for the VRF 10 instance.

```
device#show ip msdp vrf 10 mesh-group group2
MSDP MESH-GROUP:group2
KA: Keepalive SA:Source-Active NOT: Notification
Peer Address    State          KA            SA            NOT         Age
                                  In     Out    In     Out     In     Out
21.0.0.23           IDLE       0      0      0      0       0      0      0
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show ip msdp peer

Displays Multicast Source Discovery Protocol (MSDP) peer information.

## Syntax

**show ip msdp peer** [ **vrf** *vrf-name* ] **peer** *peer-address*

## Parameters

**vrf** *vrf-name*

Displays information for a specific VRF instance.

*peer-address*

Displays information for the specified peer address.

## Modes

Privileged EXEC mode

## Examples

The following example shows MSDP information about the specified peer.

```
device# show ip msdp peer 10.40.40.3
MSDP Peer Status Summary
KA: Keepalive SA:Source-Active NOT: Notification
Peer Address Peer As  State         KA     SA      NOT    Age
                                  In Out  In Out  In Out
10.40.40.3   1001    ESTABLISH    62 62   0  0    0   0   7
```

# show ip msdp rpf-peer

Displays Multicast Source Discovery Protocol (MSDP) peer information for a reverse-path forwarding (RPF) peer.

## Syntax

**show ip msdp** [ **vrf** *vrf-name* ] **rpf-peer** *peer-address*

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

*peer-address*

Specifies the source address for reverse-path forwarding (RPF) check.

## Modes

Privileged EXEC mode

## Examples

The following example shows MSDP peer information for the VRF named my_vrf.

```
device#show ip msdp vrf my_vrf rpf-peer 10.40.40.2
MSDP Peer Status Summary
KA: Keepalive SA:Source-Active NOT: Notification
 Peer Address    Peer As    State           KA           SA           NOT        Age
                                          In     Out    In     Out    In    Out
10.40.40.2      1001       ESTABLISH    5569   5568   0      0      0      0      57
```

# show ip msdp sa-cache

Displays the source actives (SA) in the Multicast Source Discovery Protocol (MSDP) cache.

## Syntax

**show ip msdp** [ **vrf** *vrf-name* ] **sa-cache** [ **counts** ] [*source-address group-address* | **peer** *peer-address* { **in** | **out** } | **peer-as** *as-number* | **orig-rp** *rp-address* | **rejected** [ **rpf** | **rp-filter** | **sg-filter** ] | **self-originated** ]

## Parameters

**vrf** *vrf-name*
Displays information for a specific VRF instance.

**counts**
Displays only the count of entries.

*source-address*
Specifies the source address of the SA entry.

*group-address*
Specifies the group address of the SA entry.

**peer-as** *as-number*
Specifies the BGP any-source (AS) number of the forwarding peer.

**orig-rp** *rp-address*
Displays information for the originating reverse-path (RP) address.

**peer** *peer-address*
Displays information for the peer address.

**in**
Displays SA entries received from this peer.

**out**
Displays SA entries advertised to this peer.

**rejected**
Displays the rejected SAs.

**rpf**
Displays the RPF failure information.

**rp-filter**
Displays the RP filter failure information.

**sg-filter**
Displays the SG failure information.

**self-originated**
Displays the self-originated SAs.

# Modes

User EXEC mode

# Command Output

The **show ip msdp sa-cache** command displays the following information:

| Output Field | Description |
|---|---|
| Total | The number of entries the cache currently contains. |
| Index | The cache entry number. |
| RP | The RP through which receivers can access the group traffic from the source |
| SourceAddr | The IP address of the multicast source. |
| GroupAddr | The IP multicast group to which the source is sending information. |
| Orig Peer | The peer from which this source-active entry was received. |
| Age | The number of seconds the entry has been in the cache |

# Examples

This example shows the source actives in the MSDP cache:

```
device> show ip msdp vrf my_vrf sa-cache
 Total of 10 SA cache entries
Index  RP address (Source, Group)          Orig Peer  Age
1      2.2.2.2    (192.6.1.10, 227.1.1.1)  192.1.1.2  0
2      2.2.2.2    (192.6.1.10, 227.1.1.2)  192.1.1.2  0
3      2.2.2.2    (192.6.1.10, 227.1.1.3)  192.1.1.2  0
4      2.2.2.2    (192.6.1.10, 227.1.1.4)  192.1.1.2  0
5      2.2.2.2    (192.6.1.10, 227.1.1.5)  192.1.1.2  0
6      2.2.2.2    (192.6.1.10, 227.1.1.6)  192.1.1.2  0
7      2.2.2.2    (192.6.1.10, 227.1.1.7)  192.1.1.2  0
8      2.2.2.2    (192.6.1.10, 227.1.1.8)  192.1.1.2  0
9      2.2.2.2    (192.6.1.10, 227.1.1.9)  192.1.1.2  0
10     2.2.2.2    (192.6.1.10, 227.1.1.10) 192.1.1.2  0
```

The following example configures to display only the entries matching a specific source.

```
device> show ip msdp sa-cache 1.1.1.1
```

The following example configures to display only the entries matching a specific group.

```
device> show ip msdp sa-cache 239.1.1.1
```

The following example configures to display only the SA cache entries that are received from peers in the BGP AS Number 100.

```
device> show ip msdp sa-cache 100
```

The following example configures to display only the SA cache entries that are originated by the RP 10.1.1.1.

```
device> show ip msdp sa-cache orig-rp 10.1.1.1
```

The following example configures to display only the rejected SAs. You can further narrow down by quoting the reason for rejection.

```
device> show ip msdp sa-cache rejected
```

The following example configures to display the self-originated SA.

```
device> show ip msdp sa-cache self-originated
```

# show ip msdp summary

Displays the IP addresses of the Multicast Source Discovery Protocol (MSDP) peers, the state of the device MSDP session with each peer, and statistics for keepalive, source active, and notification messages sent to and received from each of the peers.

## Syntax

**show ip msdp** [ **vrf** *vrf-name* ] **summary**

## Parameters

**vrf** *vrf-name*
Specifies information for a VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ip msdp summary** command displays the following information:

| Output Field | Description |
|---|---|
| Peer address | The IP address of the peer interface with the device |
| State | The state of the MSDP device connection with the peer. The state can be one of the following:<br>• CONNECTING - The session is in the active open state.<br>• ESTABLISHED - The MSDP session is fully up.<br>• INACTIVE - The session is idle.<br>• LISTENING - The session is in the passive open state. |
| KA In | The number of MSDP keepalive messages the MSDP device has received from the peer |
| KA Out | The number of MSDP keepalive messages the MSDP device has sent to the peer |
| SA In | The number of source active messages the MSDP device has received from the peer |
| SA Out | The number of source active messages the MSDP device has sent to the peer |
| NOT In | The number of notification messages the MSDP router has received from the peer |
| NOT Out | The number of notification messages the MSDP router has sent to the peer |

# Examples

The following example shows summary MSDP information for the VRF named my_vrf.

```
device# show ip msdp my_vrf summary
MSDP Peer Status Summary
KA: Keepalive SA:Source-Active NOT: Notification
Peer Address    Peer As    State          KA           SA            NOT        Age
                                       In     Out    In     Out    In     Out
40.40.40.1      1001       ESTABLISH 59     59     0      0      0      0      6
40.40.40.3      1001       ESTABLISH 59     59     0      0      0      0      47
47.1.1.2        N/A        ESTABLISH 59     59     0      0      0      0      47
```

# show ip multicast

Displays IPv4 IGMP snooping information.

## Syntax

**show ip multicast**

## Modes

User EXEC mode

## Usage Guidelines

You can use the **show ip multicast** command to display information for VLANs.

## Examples

The following example shows IGMP snooping information.

```
device# show ip multicast
Summary of all vlans. Please use "sh ip mu vlan <vlan-id>" for details
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=255,
                      Leave Wait=2, Robustness=2

Replication resource sharing: Enabled.
VL20: dft V2, vlan cfg active, 0 grp, 0 (*G) cache, rtr ports,
    router ports: e1/1/5(220) 1.1.1.20,
My Query address: None
Vlan Querier address not configured. Ve/Loopback address also not available.
VL30: dft V2, vlan cfg active, 0 grp, 0 (*G) cache, no rtr port,
Vlan Querier address configured: 30.1.1.1
VL40: dft V2, vlan cfg passive, 0 grp, 0 (*G) cache, no rtr port,
Vlan Querier address not configured. Ve/Loopback address also not available.
VL120 no snoop: no global or local config
VL200 no snoop: no global or local config
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | The output of this command was modified to display the robustness variable, leave-wait timer, and the My Query address field. |
| 08.0.30 | This command was modified to display information for unregistered flooding. |

# show ip multicast error

Displays information about possible IGMP errors.

## Syntax

**show ip multicast error**

## Modes

User EXEC mode

## Command Output

The **show ip multicast error** command displays the following information:

| Output Field | Description |
|---|---|
| SW processed pkt | The number of multicast packets processed by IGMP snooping. |
| up-time | The time since the IGMP snooping is enabled. |

## Examples

The following example shows information about possible IGMP errors.

```
device> show ip multicast error
snoop SW processed pkt: 173, up-time 160 sec
```

# show ip multicast group

Displays information about IGMP groups.

## Syntax

**show ip multicast** [ **cluster** ] **group** [ *group-address* [ **detail** ] [ **tracking** ] ]

## Parameters

**cluster**

Specifies a multi-chassis trunking (MCT) cluster.

*group-address*

Specifies information for a particular group.

**detail**

Specifies detailed IGMP group information for a specific group.

**tracking**

Specifies tracking information on interfaces that have tracking enabled.

## Modes

Privileged EXEC mode

## Command Output

The **show ip multicast group** command displays the following information:

| Output Field | Description |
|---|---|
| group | The address of the group (destination address in this case, 224.1.1.1) |
| p-port | The physical port on which the group membership was received. |
| ST | **Yes** indicates that the IGMP group was configured as a static group; **No** means the address was learned from reports. |
| QR | **Yes** means the port is a querier port; No means it is not. A port becomes a non-querier port when it receives a query from a source with a lower source IP address than the device. |
| life | The number of seconds the group can remain in EXCLUDE mode. An EXCLUDE mode changes to INCLUDE mode if it does not receive an "IS_EX" or "TO_EX" message during a certain period of time. The default is 260 seconds. There is no life displayed in INCLUDE mode. |
| mode | Indicates current mode of the interface: INCLUDE or EXCLUDE. If the interface is in INCLUDE mode, it admits traffic only from the source list. If an interface is in EXCLUDE mode, it denies traffic from the source list and accepts the rest. |
| source | Identifies the source list that will be included or excluded on the interface. |

| Output Field | Description |
|---|---|
| | For example, if an IGMP V2 group is in EXCLUDE mode with a source of 0, the group excludes traffic from the 0 (zero) source list, which actually means that all traffic sources are included. |

# Examples

The following example shows that an IGMP V2 group is in EXCLUDE mode with a source of 0. The group excludes only traffic from the 0 (zero) source list, which means that all traffic sources are included.

```
Device# show ip multicast group
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL70 : 3 groups, 4 group-port, tracking_enabled
      group           p-port   ST     QR     life mode    source
1     224.1.1.2       1/1/33   no     yes    120  EX      0
2     224.1.1.1       1/1/33   no     yes    120  EX      0
3     226.1.1.1       1/1/35   yes    yes    100  EX      0
4     226.1.1.1       1/1/33   yes    yes    100  EX      0
```

The following example displays detailed IGMP group information for multicast group 226.1.1.1:

```
Device# show ip multicast group 226.1.1.1 detail
Display group 226.1.1.1 in all interfaces in details.
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL70 : 1 groups, 2 group-port, tracking_enabled
      group           p-port   ST     QR     life mode    source
1     226.1.1.1       1/1/35   yes    yes    120  EX      0
    group: 226.1.1.1, EX, permit 0 (source, life):
      life=120, deny 0:
      group           p-port   ST     QR     life mode    source
2     226.1.1.1       1/1/33   yes    yes    120  EX      0
    group: 226.1.1.1, EX, permit 0 (source, life):
      life=120, deny 0:
```

The following example displays the list of clients that belong to multicast group 224.1.1.1 when tracking and fast leave are enabled:

```
Device# show ip multicast group 224.1.1.1 tracking
Display group 224.1.1.1 in all interfaces with tracking enabled.
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL70 : 1 groups, 1 group-port, tracking_enabled
      group           p-port   ST     QR     life mode    source
*** Note: has 1 static groups to the entire vlan, not displayed here
1     224.1.1.1       1/1/33   no     yes    100  EX      0
    receive reports from 1 clients: (age)
      (10.2.100.2 60)
```

The following example displays information for a device in an MCT cluster, In the “local” column, YES indicates that report/leave were received on local ports [cluster-edge ports (CEP) or cluster-client-edge ports (CCEP)]; NO indicates that report/leave were received on a port that is an inter-chassis link (ICL) between the MCT cluster switches, via an MCT peer.

```
Device#show ip multicast cluster group
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL70 : 1 groups, 1 group-port
      group           p-port   ST     QR     life  mode    source     local
1     225.1.1.1       e1/3/10  no     no     260   EX      0          YES
2     230.1.1.2       e1/3/12  no     yes    40    EX      0          NO
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was modified to display MCT cluster information. |

# show ip multicast mcache

Displays information in the multicast forwarding mcache.

## Syntax

**show ip multicast** [ **cluster** ] **mcache**

## Parameters

**cluster**

Specifies a multi-chassis trunking (MCT) cluster.

## Modes

Privileged EXEC mode

## Usage Guidelines

Configuring the **show default values** command does not show complete output; it shows only IGMP mcache values. The IGMP snooping mcache contains multicast forwarding information for VLANs and you must configure the **show ip multicast mcache** command to display those.

## Command Output

The **show ip multicast mcache** command displays the following information:

| Field | Description |
|---|---|
| (source group) | Source and group addresses of this data stream. (* group) means match group only; (source group) means match both. |
| cnt | The number of packets processed in software. Packets are switched in hardware, which increases this number slowly. |
| OIF | The output interfaces. If `entire vlan` is displayed, this indicates that static groups apply to the entire VLAN. |
| age | The mcache age. The mcache will be reset to 0 if traffic continues to arrive, otherwise the mcache will be aged out when it reaches the time defined by the **ip multicast mcache-age** command. |
| uptime | The up time of this mcache in seconds. |
| vidx | Vidx specifies output port list index. Range is from 4096 through 8191. |
| ref-cnt | The vidx is shared among mcaches having the same output interfaces. Ref-cnt indicates the number of mcaches using this vidx. |
| ICL | Inter-chassis link between MCT cluster switches. |
| CCEP | Cluster-client-edge ports (ports on cluster switch connecting it with acluster client). |

# Examples

The following example shows information in the multicast forwarding mcache:

```
Device#show ip multicast mcache
Example: (S G) cnt=: cnt is number of SW processed packets
    OIF: e1/1/22 TR(1/1/32,1/1/33), TR is trunk, e1/1/32 primary, e1/1/33 output
vlan 10, 1 caches. use 1 VIDX
1    (10.10.10.2 239.0.0.3) cnt=0
     OIF: tag e2
     age=2s up-time=2s change=2s vidx=8191 (ref-cnt=1)
```

The following example shows information in the multicast forwarding mcache when data arrives locally:

```
Device#show ip multicast cluster mcache
xample: (S G) cnt=: (S G) are the lowest 32 bits, cnt is number of SW processed packets
    OIF: e1/1/22 TR(e1/1/32,e1/1/33), TR is trunk, e1/1/32 primary, e1/1/33 output
         [1,10]: [1 - has local oif, 10 - ICL due to CCEP count]

vlan 10, 1 caches. use 1 VIDX
1    (* 225.1.1.3) cnt=52244
     OIF: tag TR(e1/4/23) [1,0]
     age=167s up-time=11548s, change=58639s vidx=8184 (ref-cnt=1)
```

The following example shows information in the multicast forwarding mcache when data arrives on an MCT peer:

```
Device#show ip multicast cluster mcache
Example: (S G) cnt=: (S G) are the lowest 32 bits, cnt is number of SW processed packets
    OIF: e1/1/22 TR(e1/1/32,e1/1/33), TR is trunk, e1/1/32 primary, e1/1/33 output
         [1,10]: [1 - has local oif, 10 - ICL due to CCEP count]

vlan 10, 1 caches. use 1 VIDX
1    (30.0.0.10 225.1.1.3) cnt=30084
     OIF: tag TR(e1/3/13) [1,0]
     age=152s up-time=13728s, change=9990s vidx=8184 (ref-cnt=1)
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was modified to display MCT cluster information. |

# show ip multicast optimization

Displays Internet Group Management Protocol (IGMP) snooping hardware resource-sharing information.

## Syntax

**show ip multicast optimization** [ *ipmc-num* ]

## Parameters

*ipmc-num*

Specifies the IP multicast (IPMC) group index number.

## Modes

Privileged EXEC mode

VLAN configuration mode

## Usage Guidelines

The **show ip multicast optimization** command is available only on the ICX 7250, ICX 7450, and ICX 7750 devices.

Use this command to display the availability of IPMC group indexes in the hardware and how they are used and shared.

The IPMC group index range varies depending on the platform; values out of range are not displayed.

## Examples

The following example displays resource information showing that IPMC group index 4 is shared by two users and the ports included in the set are 1/1/6 and 1/1/1:

```
Device(config)#vlan 150
Device(config-vlan-150)#show ip multicast optimization
Total IPMCs Allocated:    0; Available: 8192; Failed:    0
Index    IPMC         SetId         Users         Set
   1.    4             0x161fcbd8        2 {<1/1/6>,<1/1/1>,}
   2.    1             0x161d0930       10 {<1/1/6>,<1/1/4>,<1/1/3>,<1/1/2>,
                                           <1/1/1>,}
Sharability Coefficient:  76%
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# show ip multicast pimsm-snooping

Displays information related to PIM sparse mode (SM) snooping on the mcache.

## Syntax

**show ip multicast pimsm-snooping** [ **vlan** *vlan-id* ] [ **cache** *ip-address* ] [ **resources** ]

## Parameters

**cache** *ip-address*
Specifies the PIM SM Snooping cache.

**vlan** *vlan-id*
Specifies snooping for a VLAN.

**resources**
Specifies PIM SM snooping resources.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **show ip multicast pimsm-snooping** command to display information related to the PIM SM snooping on the outgoing interface (OIF) in the mcache.

## Examples

The following example shows PIM SM information for the mcache:

```
Device#show ip multicast pimsm-snooping
Example: Port: 1/7/3 (ref_count=1)
         ref_count: no of entries in pimsm snoop cache added this oif)

vlan 503, has 1 caches.
1    (* 225.1.1.1) has 3 pim join ports out of 4 OIF
     4/23 (ref_count=2), 4/13 (ref_count=1), 4/5 (ref_count=3),
```

# show ip multicast resource

Displays information about the software resources used.

## Syntax

**show ip multicast resource**

## Modes

User EXEC mode

## Command Output

The **show ip multicast resource** command displays the following information:

| Output Field | Description |
|---|---|
| alloc | The allocated number of units. |
| in-use | The number of units currently being used. |
| avail | The number of available units. |
| get-fail | Thie number of resource failures.<br>**NOTE**<br>It is important to pay attention to this field. |
| limit | The upper limit of this expandable field. The limit of `multicast group` is configured by the **system-max igmp-snoop-group-addr** command. The limit of `snoop mcache entry` is configured by the **system-max igmp-snoop-mcache** command. |
| get-mem | The number of memory allocation. This number must continue to increase. |
| size | The size of a unit (in bytes). |
| init | The initial allocated amount of memory. More memory may be allocated if resources run out. |
| Available vidx | The output interface (OIF) port mask used by mcache. The entire device has a maximum of 4096 vidx. Different mcaches with the same OIF share the same vidx. If vidx is not available, the stream cannot be hardware-switched. |

## Examples

The following example shows information about the software resources.

```
Device#show ip multicast resource
                     alloc in-use  avail get-fail    limit  get-mem  size init
igmp group             256      1    255        0    32000        1    16  256
igmp phy port         1024      1   1023        0   200000        1    22 1024
.... entries deleted ...
snoop mcache entry     128      2    126        0     8192        3    56  128
total pool memory 109056 bytes
has total 2 forwarding hash
VIDX sharing hash   : size=2      anchor=997  2nd-hash=no  fast-trav=no
Available vidx: 4060. IGMP/MLD use 2
```

# show ip multicast vlan

Displays IGMP snooping information for a specific VLAN.

## Syntax

**show ip multicast vlan** [ **cluster** ] *vlan-id*

## Parameters

**cluster**

Specifies a Multi-Chassis Trunking (MCT) cluster.

*vlan-id*

Specifies the VLAN for which you want information. If you do not specify a *vlan-id*, information for all VLANs is displayed.

## Modes

Privileged EXEC mode

## Usage Guidelines

You can use the **show ip multicast vlan** command to display the querier information for a VLAN. This command displays the VLAN interface status and whether there is any other querier present with the lowest IP address. The following list provides the combinations of querier possibilities:

- Active Interface with no other querier present
- Passive Interface with no other querier present
- Active Interface with other querier present
- Passive Interface with other querier present

## Command Output

The **show ip multicast vlan** command displays the following information:

| Output Field | Description |
|---|---|
| Version | The global IGMP version. |
| Query | How often a querier sends a general query on the interface. |
| Group Age | The number of seconds membership groups can be members of this group before aging out. |
| Max Resp | The maximum number of seconds a client waits before replying to a query. |
| Other Qr | How long it took a switch with a lower IP address to become a new querier. This value is 2 x Query + Max Resp. |
| Unregistered IPv4 Multicast Packets Flooding | Indicates whether flooding is enabled. |
| cfg | The IGMP version for the specified VLAN. |
| vlan cfg | The IGMP configuration mode, which is either passive or active. |

| Output Field | Description |
|---|---|
| pimsm | Indicates that PIM SM is enabled on the VLAN. |
| rtr port | The router ports, which are the ports receiving queries. |
| local | Entries learned on local interfaces of the cluster switch, for example, the local client edge port (CCEP) or cluster edge port (CEP). |
| mct peer | Entries learned by way of the MCT peer cluster switch. Control messages synchronize by way of the inter-chassis link (ICL) from the MCT peer cluster switch. |

# Examples

The following example shows IGMP snooping information for VLAN 10:

```
device# show ip multicast vlan 10
Version=3, Intervals: Query=10, Group Age=260, Max Resp=10, Other Qr=30
VL10: cfg V3, vlan cfg passive, , pimsm (vlan cfg), 3 grp, 1 (SG) cache, no rtr port,
  My Query address: None
  e2      has    3 groups, non-QR (passive), default V3
  **** Warning! has V2 client (life=240),
    group: 239.0.0.3, life = 240
    group: 224.1.1.2, life = 240
    group: 224.1.1.1, life = 240
  e4      has    0 groups, non-QR (passive), default V3
```

The following example shows IGMP snooping information when the VLAN interface is active and no other querier is present with the lowest IP address:

```
device# show ip multicast vlan 10
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=260
VL10: dft
 V2, vlan cfg active, 0 grp, 0 (*G) cache, no rtr port,
  My Query address: None
  1/1/16  has    0 groups,
This interface is Querier
default V2
  1/1/24  has    0 groups,
This interface is Querier
default V2
  2/1/16  has    0 groups,
This interface is Querier
default V2
  2/1/24  has    0 groups,
This interface is Querier
default V2
  3/1/1   has    0 groups,
This interface is Querier
default V2
  3/1/4   has    0 groups,
This interface is Querier
default V2
```

The following example shows IGMP snooping information when the VLAN interface is passive and no other querier is present with the lowest IP address:

```
device# show ip multicast vlan 10
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=260
VL10: dft V2, vlan cfg passive, 0 grp, 0 (*G) cache, no rtr port,
  My Query address: None
  1/1/16  has    0 groups,
This interface is non-Querier (passive)
default V2
  1/1/24  has    0 groups,
This interface is non-Querier (passive)
default V2
  2/1/16  has    0 groups,
This interface is non-Querier (passive)
default V2
  2/1/24  has    0 groups,
This interface is non-Querier (passive)
default V2
  3/1/1   has    0 groups,
This interface is non-Querier (passive)
default V2
  3/1/4   has    0 groups,
This interface is non-Querier (passive)
default V2
```

The following example shows IGMP snooping information when the VLAN interface is active and another querier is present with the lowest IP address:

```
device# show ip multicast vlan 10
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=260
VL10: dft V2, vlan cfg active, 7 grp, 6 (*G) cache, rtr ports,
  My Query address: None
    router ports: 2/1/24(260) 10.5.5.5, 3/1/4(260) 10.8.8.8,
  1/1/16  has    4 groups,
This interface is Querier
default V2
    group: 226.6.6.6, life = 240
    group: 228.8.8.8, life = 240
    group: 230.0.0.0, life = 240
    group: 224.4.4.4, life = 240
  1/1/24  has    1 groups,
This interface is Querier
default V2
    group: 228.8.8.8, life = 240
  2/1/16  has    4 groups,
This interface is Querier
default V2
    group: 226.6.6.6, life = 240
    group: 228.8.8.8, life = 240
    group: 230.0.0.0, life = 240
    group: 224.4.4.4, life = 240
  2/1/24  has    2 groups,
This interface is non-Querier
Querier is 10.5.5.5
Age is 0
Max response time is 100
default V2
  **** Warning! has V3 (age=0)  nbrs
    group: 234.4.4.4, life = 260
    group: 226.6.6.6, life = 260
  3/1/1   has    4 groups,
This interface is Querier
default V2
    group: 238.8.8.8, life = 260
    group: 228.8.8.8, life = 260
    group: 230.0.0.0, life = 260
    group: 224.4.4.4, life = 260
  3/1/4   has    1 groups,
This interface is non-Querier
Querier is 10.8.8.8
Age is 0
Max response time is 100
default V2
  **** Warning! has V3 (age=0)  nbrs
    group: 236.6.6.6, life = 260
```

The following example shows IGMP snooping information when the VLAN interface is passive and another querier is present with the lowest IP address:

```
device# show ip multicast vlan 10
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=260
VL10: dft V2, vlan cfg passive, 7 grp, 6 (*G) cache, rtr ports,
    router ports: 2/1/24(260) 10.5.5.5, 3/1/4(260) 10.8.8.8,
  My Query address: None
  1/1/16  has    4 groups,
This interface is non-Querier (passive)
default V2
    group: 226.6.6.6, life = 260
    group: 228.8.8.8, life = 260
    group: 230.0.0.0, life = 260
    group: 224.4.4.4, life = 260
  1/1/24  has    1 groups,
This interface is non-Querier (passive)
default V2
    group: 228.8.8.8, life = 260
  2/1/16  has    4 groups,
This interface is non-Querier (passive)
default V2
    group: 226.6.6.6, life = 260
    group: 228.8.8.8, life = 260
    group: 230.0.0.0, life = 260
    group: 224.4.4.4, life = 260
  2/1/24  has    2 groups,
This interface is non-Querier (passive)
Querier is 10.5.5.5
Age is 0
Max response time is 100
default V2
  **** Warning! has V3 (age=0)  nbrs
    group: 234.4.4.4, life = 260
    group: 226.6.6.6, life = 260
  3/1/1   has    4 groups,
This interface is non-Querier (passive)
default V2
    group: 238.8.8.8, life = 260
    group: 228.8.8.8, life = 260
    group: 230.0.0.0, life = 260
    group: 224.4.4.4, life = 260
  3/1/4   has    1 groups,
This interface is non-Querier (passive)
Querier is 10.8.8.8
Age is 0
Max response time is 100
default V2
  **** Warning! has V3 (age=0)  nbrs
    group: 236.6.6.6, life = 260
```

The following example shows IGMP snooping information when the device is connected to an MCT cluster:

```
device# show ip multicast cluster vlan 10
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=255

VL10: dft V2, vlan cfg passive, 0 grp, 0 (*G) cache, rtr ports,
  My Query address: None
    router ports: e4/14(65) 50.0.0.1 (local:1, mct peer:0)

(local:1, mct peer:0)     <- Indicates if entry is local or\and mct-peer entry
```

The following example shows IGMP snooping information when flooding of unregistered IPv4 multicast frames is disabled:

```
device#show ip multicast vlan
Summary of all vlans. Please use "sh ip mu vlan <vlan-id>" for details
Version=2, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=255

Unregistered IPv4 Multicast Packets Flooding: Disabled.

VL500: dft V2, vlan cfg active, 0 grp, 0 (*G) cache, no rtr port,
VL600 no snoop: no global or local config
```

# History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was modified to display MCT cluster information. |
| 8.0.30 | This command was modified to display flooding information. |
| 8.0.50 | The output of this command was updated to include the My Query address field. |

# show ip ospf

Displays OSPF information.

## Syntax

**show ip ospf**

## Modes

User EXEC mode

## Examples

The following example displays sample output from the **show ip ospf** command.

```
device> show ip ospf

OSPF Version                 Version 2
Router Id                    10.11.12.13
ASBR Status                  Yes
ABR Status                   No          (0)
Redistribute Ext Routes from Default-Info
Initial SPF schedule delay   100         (msecs)
Minimum hold time for SPFs   100         (msecs)
Maximum hold time for SPFs   100         (msecs)
External LSA Counter         1
External LSA Checksum Sum    00006777
Originate New LSA Counter    5033
Rx New LSA Counter           1
External LSA Limit           22
Database Overflow Interval   0
Database Overflow State :    NOT OVERFLOWED
RFC 1583 Compatibility :     Enabled
NSSA Translator:             Enabled
Nonstop Routing:             Disabled
Graceful Restart:            Enabled,   timer 120
Graceful Restart Helper:     Enabled
```

# show ip ospf area

Displays the OSPF area table in a specified format.

## Syntax

**show ip ospf area** { *A.B.C.D* | *decimal* } **database link-state** [ **advertise** *index* | **asbr** { *asbr-id* | **adv-router** *router-id* } | **extensive** | **link-state-id** *id* | **network** { *net-id* | **adv-router** *router-id* } | **nssa** { *nssa-id* | **adv-router** *router-id* } | **router** { *router-id* | **adv-router** *router-id* } | **self-originate** | **sequence-number** *num* | **summary** { *id* | **adv-router** *router-id* } ]

## Parameters

*A.B.C.D*
Area address in dotted decimal format.

*decimal*
Area address in decimal format. Valid values range from 0 to 2147483647.

**database link-state**
Displays database link-state information.

**advertise** *index*
Displays the link state by Link State Advertisement (LSA) index.

**asbr**
Displays the link state for all autonomous system boundary router (ASBR) links.

*asbr-id*
Displays the state of a single ASBR link that you specify.

**adv-router** *router-id*
Displays the link state for the advertising router that you specify.

**extensive**
Displays detailed information for all entries in the OSPF database.

**link-state-id** *id*
Displays the link state by link-state ID.

**network**
Displays the link state by network link.

*net-id*
Displays the link state of a particular network link that you specify.

**nssa**
Displays the link state by not-so-stubby area (NSSA).

*nssa-id*
Displays the link state of a particular NSAA area that you specify.

**router**
Displays the link state by router link.

*router-id*

Displays the link state of a particular router link that you specify.

**self-originate**

Displays self-originated link states.

**sequence-number** *num*

Displays the link-state by sequence number that you specify.

**summary**

Displays the link state summary. Can specify link-state ID or advertising router ID.

*id*

Displays the link state for the advertising router that you specify.

# Modes

User EXEC mode

# Command Output

The **show ip ospf area** command displays the following information:

| Output field | Description |
|---|---|
| Index | The row number of the entry in the router's OSPF area table. |
| Area | The area number. |
| Type | The area type, which can be one of the following:<br>• nssa<br>• normal<br>• stub |
| Cost | The area's cost. |
| SPFR | The SPFR value. |
| ABR | The ABR number. |
| ASBR | The ABSR number. |
| LSA | The LSA number. |
| Chksum(Hex) | The checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The device uses the checksum to verify that the packet is not corrupted. |

# Examples

The following example shows output for the **show ip ospf area** command.

```
device> show ip ospf area

Indx  Area        Type    Cost  SPFR  ABR  ASBR  LSA  Chksum(Hex)
1    0.0.0.0      normal  0     1     0    0     1    0000781f
2   10.147.60.0   normal  0     1     0    0     1    0000fee6
3   10.147.80.0   stub    1     1     0    0     2    000181cd
```

# show ip ospf border-routers

Displays information about border routers and boundary routers.

## Syntax

**show ip ospf border-routers** [ *A.B.C.D* ]

## Parameters

*A.B.C.D*

Specifies the router ID in dotted decimal format.

## Modes

User EXEC mode

## Usage Guidelines

Use this command to display information about area border routers (ABRs) and autonomous system boundary routers (ASBRs). You can display information for all ABRs and ASBRs or for a specific router.

## Command Output

The **show ip ospf border-routers** command displays the following information:

| Output field | Description |
|---|---|
| (Index) | Displayed index number of the border router. |
| Router ID | ID of the OSPF router |
| Router type | Type of OSPF router: ABR or ASBR |
| Next hop router | ID of the next hop router |
| Outgoing interface | ID of the interface on the router for the outgoing route. |
| Area | ID of the OSPF area to which the OSPF router belongs |

## Examples

The following is sample output for the **show ip ospf border-routers** command when no router ID is specified.

```
device> show ip ospf border-routers
      router ID       router type next hop router  outgoing interface  Area
1     10.65.12.1      ABR         10.1.49.2        v49                 0
1     10.65.12.1      ASBR        10.1.49.2        v49                 0
1     10.65.12.1      ABR         10.65.2.251      v201                65
1     10.65.12.1      ASBR        10.65.2.251      v201                65
```

The following is sample output for the **show ip ospf border-routers** command when a router ID is specified.

```
device> show ip ospf border-routers 192.168.98.111
router ID       router type next hop router outgoing interface  Area
192.168.98.111  ABR         193.213.111.111 4/3/1*8/3/1            0
```

# show ip ospf config

Displays general OSPF configuration information.

## Syntax

**show ip ospf config**

## Modes

User EXEC mode

## Command Output

The **show ip ospf config** command displays the following information:

| Output field | Description |
|---|---|
| Router OSPF | Shows whether or not the router OSPF is enabled. |
| Nonstop Routing | Shows whether or not the non-stop routing is enabled. |
| Graceful Restart | Shows whether or not the graceful restart is enabled. |
| Graceful Restart Helper | Shows whether or not the OSPF graceful restart helper mode is enabled. |
| Graceful Restart Time | Shows the maximum restart wait time advertised to neighbors. |
| Graceful Restart Notify Time | Shows the graceful restart notification time. |
| Redistribution | Shows whether or not the redistribution is enabled. |
| Default OSPF Metric | Shows the default OSPF metric value. |
| OSPF Auto-cost Reference Bandwidth | Shows whether or not the auto-cost reference bandwidth option is enabled. |
| Default Passive Interface | Shows whether or not the default passive interface state is enabled. |
| OSPF Redistribution Metric | Shows the OSPF redistribution metric type, which can be one of the following:<br>• Type1<br>• Type2 |
| OSPF External LSA Limit | Shows the external LSA limit value. |
| OSPF Database Overflow Interval | Shows the database overflow interval value. |
| RFC 1583 Compatibility | Shows whether or not the RFC 1583 compatibility is enabled. |
| Router id | Shows the ID of the OSPF router. |
| OSPF traps | Shows whether or not the following OSPF traps generation is enabled.<br>• Interface State Change Trap<br>• Virtual Interface State Change Trap<br>• Neighbor State Change Trap<br>• Virtual Neighbor State Change Trap<br>• Interface Configuration Error Trap<br>• Virtual Interface Configuration Error Trap |

| Output field | Description |
|---|---|
| | • Interface Authentication Failure Trap<br>• Virtual Interface Authentication Failure Trap<br>• Interface Receive Bad Packet Trap<br>• Virtual Interface Receive Bad Packet Trap<br>• Interface Retransmit Packet Trap<br>• Virtual Interface Retransmit Packet Trap<br>• Originate LSA Trap<br>• Originate MaxAge LSA Trap<br>• Link State Database Overflow Trap<br>• Link State Database Approaching Overflow Trap |
| Area-ID | Shows the area ID of the interface. |
| Area-Type | Shows the area type, which can be one of the following:<br>• nssa<br>• normal<br>• stub |
| Cost | Shows the cost of the area. |
| Ethernet Interface | Shows the OSPF interface. |
| ip ospf md5-authentication-key-activation-wait-time | Shows the wait time of the device until placing a new MD5 key into effect. |
| ip ospf area | Shows the area of the interface. |
| ip ospf cost | Shows the overhead required to send a packet across an interface. |

# Examples

The following example displays general OSPF configuration information.

```
device> show ip ospf config
Router OSPF: Enabled
Nonstop Routing: Disabled
Graceful Restart: Disabled
Graceful Restart Helper: Enabled
Graceful Restart Time: 120
Graceful Restart Notify Time: 0
Redistribution: Disabled
Default OSPF Metric: 50
OSPF Auto-cost Reference Bandwidth: Disabled
Default Passive Interface: Enabled
OSPF Redistribution Metric: Type2
OSPF External LSA Limit: 1447047
OSPF Database Overflow Interval: 0
RFC 1583 Compatibility: Enabled
Router id: 10.95.11.128
Interface State Change Trap:                        Enabled
Virtual Interface State Change Trap:                Enabled
Neighbor State Change Trap:                         Enabled
Virtual Neighbor State Change Trap:                 Enabled
Interface Configuration Error Trap:                 Enabled
Virtual Interface Configuration Error Trap:         Enabled
Interface Authentication Failure Trap:              Enabled
Virtual Interface Authentication Failure Trap:      Enabled
Interface Receive Bad Packet Trap:                  Enabled
Virtual Interface Receive Bad Packet Trap:          Enabled
Interface Retransmit Packet Trap:                   Disabled
Virtual Interface Retransmit Packet Trap:           Disabled
Originate LSA Trap:                                 Disabled
Originate MaxAge LSA Trap:                          Disabled
Link State Database Overflow Trap:                  Disabled
Link State Database Approaching Overflow Trap:      Disabled
OSPF Area currently defined:
Area-ID          Area-Type Cost
0                normal    0
OSPF Interfaces currently defined:
Ethernet Interface: 1/3/1-1/3/2
ip ospf md5-authentication-key-activation-wait-time 300
ip ospf cost 0
ip ospf area 0
Ethernet Interface: v1
ip ospf md5-authentication-key-activation-wait-time 300
ip ospf cost 0
ip ospf area 0
```

# show ip ospf database

Shows OSPFv2 database information.

## Syntax

**show ip ospf database**

**show ip ospf database database-summary**

**show ip ospf database external-link-state** [ **advertise** *index* | **extensive** | **link-state-id** *id* | **router-id** *router-id* | **sequence-number** *num* ]

**show ip ospf database grace-link-state**

**show ip ospf database link-state** [ **advertise** *index* | **asbr** [ *asbr-id* | **adv-router** *router-id* ] | **extensive** | **link-state-id** *id* | **network** { *net-id* | **adv-router** *router-id* } | **nssa** { *nssa-id* | **adv-router** *router-id* } | **router** { *router-id* | **adv-router** *router-id* } | **router-id** *router-id* | **self-originate** | **sequence-number** *num* | **summary** { *id* | **adv-router** *router-id* } ]

## Parameters

**database-summary**

Displays how many link state advertisements (LSAs) of each type exist for each area, as well as total number of LSAs.

**external-link-state**

Displays information by external link state, based on the following parameters:

**advertise** *index*

Displays the hexadecimal data in the specified LSA packet. The *index* parameter identifies the LSA packet by its position in the router's External LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf external-link-state** command.

**extensive**

Displays LSAs in decrypt format. Do not use this parameter in combination with other display parameters because the entire database is displayed.

**link-state-id** *id*

Displays external LSAs for the LSA source that you specify.

**router-id** *router-id*

Displays external LSAs for the advertising router that you specify.

**sequence-number** *num*

Displays the External LSA entries for the hexadecimal LSA sequence number that you specify.

**link-state**

Displays the link state, based on the following parameters:

**adv-router** *router-id*

Displays the link state for the advertising router that you specify.

**advertise** *index*
Displays the hexadecimal data in the specified LSA packet. The *index* parameter identifies the LSA packet by its position in the router's external-LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf external-link-state** command.

**asbr**
Displays autonomous system boundary router (ASBR) LSAs.

**extensive**
Displays LSAs in decrypt format. Do not use this parameter in combination with other display parameters because the entire database is displayed.

**link-state-id** *id*
Displays LSAs for the LSA source that you specify.

**network**
Displays either all network LSAs or the LSAs for a network that you specify.

**nssa**
Displays either all NSSA LSAs or the LSAs for a not-so-stubby area (NSSA) that you specify.

**router**
Displays LSAs by router link.

**router-id** *router-id*
Displays LSAs for the advertising router that you specify.

**self-originate**
Displays self-originated LSAs.

**sequence-number**
Displays the LSA entries for the hexadecimal LSA sequence number that you specify.

**summary**
Displays summary information. You can specify link-state ID or advertising router ID.

**adv-router** *router-id*
Displays the link state for the advertising router that you specify.

# Modes

User EXEC mode

# Command Output

The **show ip ospf database** command displays the following information:

| Output field | Description |
|---|---|
| Index | ID of the entry |
| Area ID | ID of the OSPF area |
| Type | Link state type of the route. |
| LS ID | The ID of the link-state advertisement from which the router learned this route. |
| Adv Rtr | ID of the advertised route. |

| Output field | Description |
|---|---|
| Seq(Hex) | The sequence number of the LSA. The OSPF neighbor that sent the LSA stamps the LSA with a sequence number. This number enables the device and other OSPF routers to determine which LSA for a given route is the most recent. |
| Age | The age of the LSA in seconds. |
| Chksum | The checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The device uses the checksum to verify that the packet is not corrupted. |
| SyncState | This field indicates whether the synchronization is complete or not. |

# Examples

The following example shows output for the **show ip ospf database** command.

```
device> show ip ospf database

Index Area ID              Type LS ID                        Adv Rtr                       Seq(Hex)
Age  Cksum  SyncState
1     0.0.0.200   Rtr  192.168.98.111  192.168.98.111  8000003b 626  0xf885 Done
2     0.0.0.200   Rtr  192.168.98.213  192.168.98.213  800000c9 963  0x209c Done
3     0.0.0.200   Rtr  192.168.98.113  192.168.98.113  80000028 169  0x0275 Done
4     0.0.0.200   Rtr  192.168.98.112  192.168.98.112  8000002d 226  0x1c03 Done
5     0.0.0.200   Net  193.113.111.113 192.168.98.113  8000001f 1132 0x353d Done
6     0.0.0.200   Net  192.213.111.213 192.168.98.213  8000002d 1683 0x17bc Done
```

The following example shows output for the **show ip ospf database** command when the **link-state** parameter is used.

```
device> show ip ospf database link-state

Index Area ID Type LS ID       Adv Rtr     Seq(Hex) Age  Cksum           SyncState
1     0       Rtr  10.1.10.1   10.1.10.1   800060ef 3    0x4be2          Done
2     0       Rtr  10.65.12.1  10.65.12.1  80005264 6    0xc870          Done
3     0       Net  10.1.64.2   10.65.12.1  8000008c 1088 0x06b7          Done
4     0       Net  10.1.167.2  10.65.12.1  80000093 1809 0x86c8          Done
5     0       Net  10.1.14.2   10.65.12.1  8000008c 1088 0x2ec1             Done
6     0       Net  10.1.117.2  10.65.12.1  8000008c 1087 0xbccb             Done
7     0       Net  10.1.67.2   10.65.12.1  8000008c 1088 0xe4d5          Done
8     0       Net  10.1.170.2  10.65.12.1  80000073 604  0xa5c6          Done
9     0       Net  10.1.17.2   10.65.12.1  8000008c 1088 0x0ddf          Done
10    0       Net  10.1.120.2  10.65.12.1  8000008c 1087 0x9be9          Done
11    0       Net  10.1.70.2   10.65.12.1  8000008c 1088 0xc3f3          Done
12    0       Net  10.1.173.2  10.65.12.1  80000017 1087 0x3d88          Done
13    0       Net  10.1.20.2   10.65.12.1  8000008c 1088 0xebfd          Done
14    0       Net  10.1.123.2  10.65.12.1  8000008c 1087 0x7a08          Done
15    0       Net  10.1.73.2   10.65.12.1  8000008c 1088 0xa212          Done
16    0       Net  10.1.176.2  10.65.12.1  80000025 1087 0xffb4          Done
17    0       Net  10.1.23.2   10.65.12.1  8000008c 1088 0xca1c          Done
18    0       Net  10.1.126.2  10.65.12.1  8000008c 1087 0x5926             Done
```

The following example shows output for the **show ip ospf database** command when the **external-link-state** parameter is used.

```
device> show ip ospf database external-link-state

Index Age LS ID                        Router       Netmask     Metric      Flag        Fwd Address
SyncState
1     591 10.65.13.0 10.65.12.1 ffffff00 8000000a 0000        0.0.0.0              Done
2     591 10.65.16.0 10.65.12.1 ffffff00 8000000a 0000     0.0.0.0                 Done
3     591 10.65.14.0 10.65.12.1 ffffff00 8000000a 0000     0.0.0.0                 Done
4     591 10.65.17.0 10.65.12.1    ffffff00 8000000a 0000     0.0.0.0                 Done
5     592 10.65.12.0 10.65.12.1 ffffff00 8000000a 0000     0.0.0.0                 Done
6     592 10.65.15.0 10.65.12.1 ffffff00 8000000a 0000     0.0.0.0                 Done
7     592 10.65.18.0 10.65.12.1 ffffff00 8000000a 0000     0.0.0.0              Done
```

The following example shows output for the **show ip ospf database** command when the **database-summary** parameter is used.

```
device> show ip ospf database database-summary

Area ID    Router Network Sum-Net Sum-ASBR NSSA-Ext Opq-Area Subtotal
0.0.0.0    104    184     19      42       0        0        349
AS External                                                  308
Total      104    184     19      42       0        0        657
```

# show ip ospf interface

Displays information about all or specific OSPF-enabled interfaces.

## Syntax

**show ip ospf interface** [ *ip address* ] [ **brief** ] [ **ethernet** *unit/slot/port* ] [ **loopback** *number* ] [ **tunnel** *number* ] [ **ve** *vlan_id* ]

## Parameters

*ip address*
Specifies interface IP address in dotted decimal format.

**brief**
Displays brief summary information about the specified interface.

**ethernet** *unit/slot/port*
Specifies an Ethernet interface. Specify the interface ID in the format unit/slot/port-id.

**loopback** *number*
Specifies a loopback port number in the range of 1 to 255.

**tunnel** *number*
Specifies a tunnel interface.

**ve** *vlan_id*
Specifies the VLAN number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **brief** keyword to limit the display to the following fields:

- Interface
- Area
- IP address
- Cost
- State
- Nbrs(F/C)

## Command Output

The **show ip ospf interface** command displays the following information:

| This field | Displays |
|---|---|
| Interface | The type of interface type and the port number or number of the interface. |
| IP Address | The IP address of the interface. |
| Area | The OSPF area configured on the interface |
| Database Filter | The router's configuration for blocking outbound LSAs on an OSPF interface.<br>If Not Configured is displayed, there is no outbound LSA filter configured. This is the default condition. |
| State | The state of the interface. Possible states include the following:<br>• DR - The interface is functioning as the Designated Router for OSPFv2.<br>• BDR - The interface is functioning as the Backup Designated Router for OSPFv2.<br>• Loopback - The interface is functioning as a loopback interface.<br>• P2P - The interface is functioning as a point-to-point interface.<br>• Passive - The interface is up but it does not take part in forming an adjacency.<br>• Waiting - The interface is trying to determine the identity of the BDR for the network.<br>• None - The interface does not take part in the OSPF interface state machine.<br>• Down - The interface is unusable. No protocol traffic can be sent or received on such a interface.<br>• DR other - The interface is a broadcast or NBMA network on which another router is selected to be the DR.<br>• Active - The interface sends or receives all the OSPFv2 control packets and forms the adjacency. |
| default | Shows whether or not the default passive state is set. |
| Pri | The interface priority. |
| Cost | The configured output cost for the interface. |
| Interface bandwidth | The configured bandwidth on a tunnel interface for routing metric purposes only. |
| Options | OSPF Options (Bit7 - Bit0):<br>• unused:1<br>• opaque:1<br>• summary:1<br>• dont_propagate:1<br>• nssa:1<br>• multicast:1<br>• external route capable:1<br>• tos:1 |
| Type | The area type, which can be one of the following:<br>• Broadcast<br>• Point to Point<br>• non-broadcast<br>• Virtual Link |

| This field | Displays |
|---|---|
| Events | OSPF Interface Event:<br>• Interface_Up = 0x00<br>• Wait_Timer = 0x01<br>• Backup_Seen = 0x02<br>• Neighbor_Change = 0x03<br>• Loop_Indication = 0x04<br>• Unloop_Indication = 0x05<br>• Interface_Down = 0x06<br>• Interface_Passive = 0x07 |
| Timer intervals | The interval, in seconds, of the transmit-interval, retransmit-interval, hello-interval, and dead-interval timers. |
| DR | The router ID (IPv4 address) of the DR. |
| BDR | The router ID (IPv4 address) of the BDR. |
| Neighbor Count | The number of neighbors to which the interface is connected. |
| Adjacent Neighbor Count | The number of adjacent neighbor routers. |
| Neighbor: | The IP address of the neighbor. |

# Examples

This example shows sample output from the **show ip ospf interface** command when the **brief**keyword is used.

```
device# # show ip ospf interface brief
Number of Interfaces is 1
Interface Area IP Addr/Mask Cost State Nbrs(F/C)
eth 1/1/2    0    16.1.1.2/24   1     down  0/0
```

This example displays information about a specified OSPF-enabled VE interface.

```
device# show ip ospf interface ve 20

ve 20  admin up, oper up, ospf enabled, state up
     IP Address 21.21.21.22, Area 0
     Database Filter: Not Configured
     State BDR, Pri 1, Cost 1, Options 2, Type broadcast Events 31
     Timers(sec): Transmit 1, Retrans 5, Hello 10, Dead 40
     DR:  Router ID 3.3.3.3           Interface Address 21.21.21.21
     BDR: Router ID 2.2.2.2           Interface Address 21.21.21.22
                    Packets Received        Packets Sent
     Hello                      86374               86735
     Database                       2                   4
     LSA Req                        1                   0
     LSA Upd                      451                 907
     LSA Ack                      906                 451
     No Packet Errors!
     Neighbor Count = 1, Adjacent Neighbor Count= 1
     Neighbor:         21.21.21.21 [id 3.3.3.3] (DR)
     Authentication-Key: None
     MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

This example displays information about a specified OSPF-enabled Ethernet interface, including the cost, where the cost is calculated using the default interface speed and auto cost.

```
device# show ip ospf interface ethernet 3/1/1

     e 3/1/1 admin up, oper up, ospf enabled, state up
     IP Address 89.0.0.2, Area 0
     Database Filter: Not Configured
     State BDR, Pri 1, Cost 1, Options 2, Type broadcast Events 3
```

This example displays information about a specified OSPF-enabled Ethernet interface, including the cost, which has been calculated using the configured interface bandwidth and the default auto-cost.

```
device# show ip ospf interface ethernet 1/1/3

e 1/1/3 admin up, oper up, ospf enabled, state up
     IP Address 172.201.3.2, Area 0
     Database Filter: Not Configured
     State DR, Pri 1, Cost 34, Options 2, Type broadcast Events 5
     Timers(sec): Transmit 1, Retrans 5, Hello 10, Dead 40
     DR:  Router ID 192.168.3.1       Interface Address 172.201.3.2
     BDR: Router ID 192.168.1.1       Interface Address 172.201.3.1
                    Packets Received        Packets Sent
     Hello                          73                  79
     Database                        3                   2
     LSA Req                         0                   1
     LSA Upd                         4                   5
     LSA Ack                         5                   3
     No Packet Errors!
     Neighbor Count = 1, Adjacent Neighbor Count= 1
     Neighbor:         172.201.3.1 [id 192.168.1.1] (BDR)
     Authentication-Key: None
     MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was modified to include configured bandwidth status. |

# show ip ospf neighbor

Displays OSPF neighbor information.

## Syntax

**show ip ospf neighbor** [ **extensive** | *num* | **router-id** *A.B.C.D* ]

## Parameters

**extensive**

Displays detailed neighbor information.

*num*

Specifies displays only the entry in the specified index position in the neighbor table. For example, if you enter "1", only the first entry in the table is displayed.

**router-id** *A.B.C.D*

Displays neighbor information for the specified router ID.

## Modes

User EXEC mode

## Command Output

The **show ip ospf neighbor** command displays the following information:

| Output field | Description |
|---|---|
| Port | The port through which the device is connected to the neighbor. |
| Address | The IP address of the port on which this device is connected to the neighbor. |
| Pri | The OSPF priority of the neighbor.<br>• For multi-access networks, the priority is used during election of the Designated Router (DR) and Backup designated Router (BDR).<br>• For point-to-point links, this field shows one of the following values:<br>• 1 = point-to-point link<br>• 3 = point-to-point link with assigned subnet |
| State | The state of the conversation between the device and the neighbor. This field can have one of the following values:<br>• Down - The initial state of a neighbor conversation. This value indicates that there has been no recent information received from the neighbor.<br>• Attempt - This state is only valid for neighbors attached to non-broadcast networks. It indicates that no recent information has been received from the neighbor.<br>• Init - A Hello packet has recently been seen from the neighbor. However, bidirectional communication has |

| Output field | Description |
|---|---|
| | not yet been established with the neighbor. (The router itself did not appear in the neighbor's Hello packet.) All neighbors in this state (or higher) are listed in the Hello packets sent from the associated interface.<br>• 2-Way - Communication between the two routers is bidirectional. This is the most advanced state before beginning adjacency establishment. The Designated Router and Backup Designated Router are selected from the set of neighbors in the 2-Way state or greater.<br>• ExStart - The first step in creating an adjacency between the two neighboring routers. The goal of this step is to decide which router is the master, and to decide upon the initial Database Description (DD) sequence number. Neighbor conversations in this state or greater are called adjacencies.<br>• Exchange - The router is describing its entire link state database by sending Database Description packets to the neighbor. Each Database Description packet has a DD sequence number, and is explicitly acknowledged. Only one Database Description packet can be outstanding at any time. In this state, Link State Request packets can also be sent asking for the neighbor's more recent advertisements. All adjacencies in Exchange state or greater are used by the flooding procedure. In fact, these adjacencies are fully capable of transmitting and receiving all types of OSPF routing protocol packets.<br>• Loading - Link State Request packets are sent to the neighbor asking for the more recent advertisements that have been discovered (but not yet received) in the Exchange state.<br>• Full - The neighboring routers are fully adjacent. These adjacencies will now appear in router links and network link advertisements. |
| Neigh Address | The IP address of the neighbor.<br>For point-to-point links, the value is as follows:<br>• If the **Pri** field is "1", this value is the IP address of the neighbor router's interface.<br>• If the **Pri** field is "3", this is the subnet IP address of the neighbor router's interface. |
| Neigh ID | The neighbor router's ID. |
| Ev | The number of times the neighbor's state changed. |
| Opt | The sum of the option bits in the Options field of the Hello packet. This information is used by Ruckus technical support. Refer to Section A.2 in RFC 2178 for information about the Options field in Hello packets. |
| Cnt | The number of LSAs that were retransmitted. |

# Examples

The following example displays information about OSPF neighbors.

```
device> show ip ospf neighbor

Port  Address        Pri  State     Neigh Address  Neigh ID     Ev Op Cnt
v10   10.1.10.1      1    FULL/DR   10.1.10.2      10.65.12.1    5  2  0
v11   10.1.11.1      1    FULL/DR   10.1.11.2      10.65.12.1    5  2  0
v12   10.1.12.1      1    FULL/DR   10.1.12.2      10.65.12.1    5  2  0
v13   10.1.13.1      1    FULL/DR   10.1.13.2      10.65.12.1    5  2  0
v14   10.1.14.1      1    FULL/DR   10.1.14.2      10.65.12.1    5  2  0
```

# show ip ospf redistribute route

Displays routes that have been redistributed into OSPF.

## Syntax

**show ip ospf redistribute route** [ *A.B.C.D:M* ]

## Parameters

*A.B.C.D:M*

Specifies an IP address and mask for the output.

## Modes

User EXEC mode

## Examples

The following example shows sample output for the **show ip ospf redistribute route** command when no IP address and network mask are specified.

```
device> show ip ospf redistribute route

  4.3.0.0 255.255.0.0 static
  3.1.0.0 255.255.0.0 static
  10.11.61.0 255.255.255.0 connected
  4.1.0.0 255.255.0.0 static
```

The following example shows sample output for the **show ip ospf redistribute route** command when an IP address and network mask is specified.

```
device> show ip ospf redistribute route 192.213.1.0 255.255.255.254

  192.213.1.0 255.255.255.254 fwd 0.0.0.0 (0) metric 10 connected
```

# show ip ospf routes

Displays OSPF calculated routes.

## Syntax

**show ip ospf routes** [ *A.B.C.D* ]

## Parameters

*A.B.C.D*

Specifies a destination IP address in dotted decimal format.

## Modes

User EXEC mode

## Command Output

The **show ip ospf routes** command displays the following information:

| Output field | Description |
|---|---|
| Destination | The IP address of the route's destination. |
| Mask | The network mask for the route. |
| Path_Cost | The cost of this route path. (A route can have multiple paths. Each path represents a different exit port for the device.) |
| Type2_Cost | The type 2 cost of this path. |
| Path_Type | The type of path, which can be one of the following:<br>• – Inter - The path to the destination passes into another area.<br>– Intra - The path to the destination is entirely within the local area.<br>– External1 - The path to the destination is a type 1 external route.<br>– External2 - The path to the destination is a type 2 external route. |
| Adv_Router | The OSPF router that advertised the route to this device. |
| Link-State | The link state from which the route was calculated. |
| Dest_Type | The destination type, which can be one of the following:<br>• – ABR - Area Border Router<br>– ASBR - Autonomous System Boundary Router<br>– Network - the network |
| State | The route state, which can be one of the following:<br>• – Changed<br>– Invalid<br>– Valid<br>This information is used by Ruckus technical support. |
| Tag | The external route tag. |

| Output field | Description |
|---|---|
| Flags | State information for the route entry. This information is used by Ruckus technical support. |
| Paths | The number of paths to the destination. |
| Out_Port | The router port through which the device reaches the next hop for this route path. |
| Next_Hop | The IP address of the next-hop router for this path. |
| Type | The route type, which can be one of the following:<br>• – OSPF<br>– Static Replaced by OSPF |
| State | State information for the path. This information is used by Ruckus technical support. |

# Examples

The following example displays all OSPF-calculated routes.

```
device> show ip ospf route

OSPF Area 0x00000000 ASBR Routes 1:
      Destination     Mask            Path_Cost Type2_Cost Path_Type
      10.65.12.1      255.255.255.255 1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      10.65.12.1      10.65.12.1      Asbr      Valid      0          6000
      Paths Out_Port  Next_Hop        Type      State
      1     v49       10.1.49.2       OSPF      21 01
      2     v12       10.1.12.2       OSPF      21 01
      3     v11       10.1.11.2       OSPF      21 01
      4     v10       10.1.10.2       OSPF      00 00
OSPF Area 0x00000041 ASBR Routes 1:
      Destination     Mask            Path_Cost Type2_Cost Path_Type
      10.65.12.1      255.255.255.255 1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      10.65.12.1      10.65.12.1      Asbr      Valid      0          6000
      Paths Out_Port  Next_Hop        Type      State
      1     v204      10.65.5.251     OSPF      21 01
      2     v201      10.65.2.251     OSPF      20 d1
      3     v202      10.65.3.251     OSPF      20 cd
      4     v205      10.65.6.251     OSPF      00 00
OSPF Area Summary Routes 1:
      Destination     Mask            Path_Cost Type2_Cost Path_    Type
      10.65.0.0       255.255.0.0     0         0          Inter
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      10.1.10.1       0.0.0.0         Network   Valid      0          0000
      Paths Out_Port  Next_Hop        Type      State
      1     1/1/1     0.0.0.0         DIRECT    00 00
OSPF Regular Routes 208:
      Destination     Mask            Path_Cost Type2_Cost Path_Type
      10.1.10.0       255.255.255.252 1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      10.1.10.1       10.1.10.2       Network   Valid      0          0000
      Paths Out_Port  Next_Hop        Type      State
      1     v10       0.0.0.0         OSPF      00 00
      Destination     Mask            Path_Cost Type2_Cost Path_Type
      10.1.11.0       255.255.255.252 1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      10.1.10.1       10.1.11.2       Network   Valid      0          0000
      Paths Out_Port  Next_Hop        Type      State
      1     v11       0.0.0.0         OSPF      00 00
```

# show ip ospf summary

Displays summary information for all OSPF instances.

## Syntax

**show ip ospf summary**

## Modes

User EXEC mode

## Examples

```
device> show ip ospf summary

Seq Instance        Intfs   Nbrs    Nbrs-Full LSAs    Routes
1   default-vrf     5       2       1         12      2
```

# show ip ospf traffic

Displays OSPF traffic details.

## Syntax

**show ip ospf traffic**

## Modes

User EXEC mode

## Examples

The following example shows all OSPF traffic.

```
device> show ip ospf traffic

                 Packets Received         Packets Sent
Hello                          10                   10
Database                       90                   89
LSA Req                        12                   11
LSA Upd                        12                   12
LSA Ack                        12                   12
No Packet Errors!
```

# show ip ospf trap

Displays OSPF trap status.

## Syntax

**show ip ospf trap**

## Modes

User EXEC mode

## Examples

The following example shows all OSPF traffic.

```
device> show ip ospf trap

Interface State Change Trap:                      Enabled
Virtual Interface State Change Trap:              Enabled
Neighbor State Change Trap:                       Enabled
Virtual Neighbor State Change Trap:               Enabled
Interface Configuration Error Trap:               Enabled
Virtual Interface Configuration Error Trap:       Enabled
Interface Authentication Failure Trap:            Enabled
Virtual Interface Authentication Failure Trap:    Enabled
Interface Receive Bad Packet Trap:                Enabled
Virtual Interface Receive Bad Packet Trap:        Enabled
Interface Retransmit Packet Trap:                 Disabled
Virtual Interface Retransmit Packet Trap:         Disabled
Originate LSA Trap:                               Disabled
Originate MaxAge LSA Trap:                        Disabled
Link State Database Overflow Trap:                Disabled
Link State Database Approaching Overflow Trap:    Disabled
```

# show ip ospf virtual link

Displays information about virtual links.

## Syntax

**show ip ospf virtual link** [ *index* ]

## Parameters

*index*

Shows information about all virtual links or one virtual link that you specify.

## Modes

User EXEC mode

## Examples

The following example shows information about all virtual links.

```
device> show ip ospf virtual link

Indx Transit Area     Router ID        Transit(sec) Retrans(sec) Hello(sec)
1    1                131.1.1.10       1            5            10
     Dead(sec)        events           state        Authentication-Key
     40               1                ptr2ptr      None
     MD5 Authentication-Key:       None
     MD5 Authentication-Key-Id:    None
     MD5 Authentication-Key-Activation-Wait-Time:      300
```

# show ip ospf virtual neighbor

Displays information about virtual neighbors.

## Syntax

**show ip ospf virtual neighbor** [ *index* ]

## Parameters

*index*

Shows information about all virtual neighbors or one virtual neighbor that you specify.

## Modes

User EXEC mode

## Examples

The following example shows information about all virtual neighbors.

```
device> show ip ospf virtual neighbor

Indx Transit Area    Router ID       Neighbor address options
1    1               131.1.1.10      135.14.1.10      2
     Port   Address         state           events          count
     6/2/3          27.11.1.27      FULL            5               0
```

# show ip multicast traffic

Displays status information for IGMP snooping traffic.

## Syntax

**show ip multicast traffic**

## Modes

User EXEC mode

## Command Output

The **show ip multicast traffic** command displays the following information:

| Output Field | Description |
|---|---|
| Q | Query |
| Qry | General Query |
| QryV2 | Number of general IGMP V2 queries received or sent. |
| QryV3 | Number of general IGMP V3 queries received or sent. |
| G-Qry | Number of group-specific queries received or sent. |
| GSQry | Number of group source-specific queries received or sent. |
| Mbr | The membership report. |
| MbrV2 | The IGMP V2 membership report. |
| MbrV3 | The IGMP V3 membership report. |
| IsIN | Number of source addresses that were included in the traffic. |
| IsEX | Number of source addresses that were excluded in the traffic. |
| ToIN | Number of times the interface mode changed from EXCLUDE to INCLUDE. |
| ToEX | Number of times the interface mode changed from INCLUDE to EXCLUDE. |
| ALLO | Number of times that additional source addresses were allowed on the interface. |
| BLK | Number of times that sources were removed from an interface. |
| Pkt-Err | Number of packets having errors, such as checksum. |
| Pimsm-snooping hello, join, prune | Number of PIM sparse hello, join, and prune packets |

# Examples

The following example shows information for IGMP snooping traffic.

```
device> show ip multicast traffic
IGMP snooping: Total Recv: 22, Xmit: 26
Q: query, Qry: general Q,  G-Qry: group Q,  GSQry: group-source Q, Mbr: member
Recv       QryV2      QryV3      G-Qry      GSQry      MbrV2      MbrV3      Leave
VL1            0          0          0          0          4          0          0
VL70          18          0          0          0          0          0          0
Recv        IsIN       IsEX       ToIN       ToEX      ALLOW      BLOCK    Pkt-Err
VL1            0          4          0          0          0          0          0
VL70           0          0          0          0          0          0          0
Send       QryV2      QryV3      G-Qry      GSQry      MbrV2      MbrV3
VL1            0          0          8          0          0          0
VL70           0          0          0          0          0         18
VL70   pimsm-snooping, Hello:    12,  Join/Prune:     9
```

# show ip pim bsr

Displays bootstrap router (BSR) information.

## Syntax

**show ip pim** [ **all-vrf** | **vrf** *vrf-name* ] **bsr**

## Parameters

**all-vrf**

Displays information for all VRFs.

**vrf** *vrf-name*

Displays information for a specific VRF instance.

**bsr**

Displays BSR information.

## Modes

User EXEC mode

## Usage Guidelines

When entered without the **vrf** option, this command displays information for the default VRF instance.

## Command Output

The **show ip pim bsr** command displays the following information:

| Output Field. | Description |
|---|---|
| BSR address | The IP address of the interface configured as the PIM Sparse BSR. |
| BSR priority | The priority assigned to the interface for use during the BSR election process. During BSR election, the priorities of the candidate BSRs are compared and the interface with the highest BSR priority becomes the BSR. |
| Hash mask length | The number of significant bits in the IP multicast group comparison mask. This mask determines the IP multicast group numbers for which the device can be a BSR. The default is 32 bits, which allows the device to be a BSR for any valid IP multicast group number.<br><br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| Next bootstrap message in | Indicates how much time will pass before the BSR sends the next bootstrap message. The time is displayed in "hh:mm:ss" format. |

| Output Field. | Description |
|---|---|
| | **NOTE**<br>This field appears only if this device is the BSR. |
| Next Candidate-RP-advertisement message in | Indicates how much time will pass before the BSR sends the next candidate PR advertisement message. The time is displayed in "hh:mm:ss" format.<br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| RP | Indicates the IP address of the Rendezvous Point (RP).<br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| group prefixes | Indicates the multicast groups for which the RP listed by the previous field is a candidate RP.<br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| Candidate-RP-advertisement period | Indicates how frequently the BSR sends candidate RP advertisement messages.<br>**NOTE**<br>This field appears only if this device is a candidate BSR. |

# Examples

The following example shows information for a device that has been elected as the BSR.

```
device> show ip pim bsr
PIMv2 Bootstrap information for Vrf Instance : default-vrf
------------------------------------------------------------------------------
  This system is the Elected BSR
  BSR address: 1.51.51.1. Hash Mask Length 32. Priority 255.
  Next bootstrap message in 00:01:00
  Configuration:
    Candidate loopback 2 (Address 1.51.51.1). Hash Mask Length 32. Priority 255.
Next Candidate-RP-advertisment in 00:01:00
  RP: 1.51.51.1
    group prefixes:
    224.0.0.0 / 4
  Candidate-RP-advertisement period: 60
```

The following example shows information for a device that is not the BSR.

```
device(config)# show ip pim bsr
PIMv2 Bootstrap information for Vrf Instance : default-vrf
------------------------------------------------------------------------------
  BSR address: 1.51.51.1. Hash Mask Length 32. Priority 255.
Next Candidate-RP-advertisment in 00:00:30
  RP: 1.51.51.3
    group prefixes:
    224.0.0.0 / 4
  Candidate-RP-advertisement period: 60
```

# show ip pim counter nsr

Displays multicast nonstop routing (NSR) counter and statistics information.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **counter nsr**

## Parameters

**vrf** *vrf-name*
Displays information for a VRF instance.

**counter nsr**
Displays NSR counter and statistics information.

## Modes

User EXEC mode

## Usage Guidelines

When entered without the **vrf** option, this command displays information for the default VRF instance.

## Command Output

The **show ip pim counter nsr** command displays the following information:

| Output Field | Description |
|---|---|
| Mcache sync | The mcache NSR sync queue that carries the NSR sync message for mcache updates. |
| pack | The number of NSR sync messages that are packed from the active module to the standby module. |
| unpack | The number of NSR sync messages that are received and unpacked by the standby module. |
| ack | The number of NSR sync acknowledgements received by the active module. |
| RPset sync | The RPset sync queue that carries the NSR sync message for RPset update. |
| BSR status | The BSR status sync queue that carries the NSR sync message for BSR information update. |

# Examples

The following example displays PIM NSR counter and statistic information.

```
device> show ip pim counter nsr
Mcache sync (entity id: 203)
  pack: 0
  unpack: 0
  ack: 0
RPset sync (entity id: 201)
  pack: 0
  unpack: 0
  ack: 0
BSR status (entity id: 202)
  pack: 1
  unpack: 0
  ack: 1
```

# show ip pim dense

Displays PIM Dense configuration information.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **dense**

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

**dense**

Displays PIM Dense configuration information.

## Modes

User EXEC mode

## Usage Guidelines

When entered without the **vrf** option, this command displays information for the default VRF instance.

## Command Output

The **show ip pim dense** command displays the following information:

| Field | Description |
|---|---|
| Maximum Mcache | The maximum number multicast cache entries allowed on the device. |
| Current Count | The number of multicast cache entries currently used. |
| Hello interval | How frequently the device sends hello messages out the PIM dense interfaces. |
| Neighbor timeout | The interval after which a PIM device will consider a neighbor to be absent. |
| Join/Prune interval | How long a PIM device will maintain a prune state for a forwarding entry. |
| Inactivity interval | How long a forwarding entry can remain unused before the device deletes it. |
| Hardware Drop Enabled | Displays Yes if the Passive Multicast Route Insertion feature is enabled and No if it is not. |
| Prune Wait Interval | The amount of time a PIM device waits before stopping traffic to neighbor devices that do not want the traffic. The value can be from zero to three seconds. The default is three seconds. |
| Graft Retransmit interval | The interval between the transmission of graft messages. |
| Prune Age | The number of packets the device sends using the path through the RP before switching to using the SPT path. |

| Field | Description |
|---|---|
| Route Precedence | The route precedence configured to control the selection of routes based on the four route types:<br>• Non-default route from the mRTM<br>• Default route from the mRTM<br>• Non-default route from the uRTM<br>• Default route from the uRTM |

# Examples

The following example displays PIM Dense configuration information.

```
device> show ip pim dense

Global PIM Dense Mode Settings
  Maximum Mcache            : 12992    Current Count                    : 2
  Hello interval            : 30       Neighbor timeout                 : 105
  Join/Prune interval       : 60       Inactivity interval              : 180
  Hardware Drop Enabled     : Yes      Prune Wait Interval              : 3
  Graft Retransmit interval : 180      Prune Age                        : 180
  Route Precedence          : mc-non-default mc-default uc-non-default uc-default
```

# show ip pim group

Displays PIM group information.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **group**

## Parameters

**vrf** *vrf-name*

Displays information for a VRF instance.

## Modes

User EXEC mode

## Usage Guidelines

When entered without the **vrf** option, this command displays information for the default VRF instance.

## Command Output

The **show ip pim group** command displays the following information:

| Output Field | Description |
|---|---|
| Total number of groups | Lists the total number of IP multicast groups the device is forwarding.<br><br>**NOTE**<br>This list can include groups that are not PIM Sparse groups. If interfaces on the device are configured for regular PIM (dense mode), these groups are listed too. |
| Index | The index number of the table entry in the display. |
| Group | The multicast group address |
| Ports | The device ports connected to the receivers of the groups. |

# Examples

The following example displays PIM group information.

```
device> show ip pim group
Total number of groups for VRF default-vrf: 7
1    Group 226.0.34.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
2    Group 226.0.77.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
3    Group 226.0.120.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
4    Group 226.0.163.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
5    Group 226.0.206.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
6    Group 226.0.249.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
7    Group 226.0.30.0
     Group member at  e1/2/9: v59
     Group member at  e1/1/16: v57
```

# show ip pim hw-resource

Displays usage and fail-count information for SG entries.

## Syntax

**show ip pim** { **all-vrf** | **vrf** *vrf-name* } **hw-resource**

## Parameters

**all-vrf**

Displays information for all VRF instances.

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ip pim hw-resource** command displays the following information:

| Output field | Description |
|---|---|
| VRF | Name of the VRF. |
| Usage | Number of allocated SG entries in this VRF. |
| Fail | Number of failures while allocating SG entries in this VRF (due to the system-max limit. |
| Total usage | Total number of SG entries in the system (all VRFs). |
| System-max limit for SG entries | Configured system limit for pim-hw-mcache. |

## Examples

The following sample out from the show ip pim all-vrf hw-resource command displays usage and fail-count information for SG entries on each VRF.

```
device# show ip pim all-vrf hw-resource
VRF           Usage     Fail
default-vrf   3072      8
blue          3072      0
-------------------------------
Total usage   6144

System-max limit for SG entries: 6144
```

# show ip pim interface

Displays information for PIM interfaces.

## Syntax

**show ip pim interface** { **ethernet** *unit*/*slot*/*port* | **loopback** *loopback-number* | **ve** *ve-number* }

## Parameters

**ethernet** *unit*/*slot*/*port*

Specifies a physical interface. On standalone devices specify the interface ID in the format slot/port-id; on stacked devices you must also specify the stack ID, in the format stack-id/slot/port-id.

**loopback** *loopback-number*

Specifies a loopback interface.

**ve** *ve-number*

Specifies a virtual interface.

## Modes

Privileged EXEC mode

## Examples

This example displays output from the **show ip pim interface** command, showing that ACL 10 is applied to interface 1/1/9 to control neighbor access.

```
device# show ip pim interface
Flags    : SM  - Sparse Mode v2, DM - Dense Mode v2, P - Passive Mode

--------+---------------+-----+---+-------+---+-----+-------+-------+----+--------+

Int'face|Local          |Mode |St |Des Rtr|TTL|Mcast| Filter| VRF   |DR  |Override
        |Address        |     |   |AddPort|Thr|Bndry|  ACL  |       |Prio|Interval
--------+---------------+-----+---+-------+---+-----+-------+-------+----+--------+
e1/1/1   5.5.5.5         SM   Ena  Itself   1  None   None   default  1   3000ms
e1/1/9   15.1.1.5        SM   Ena  Itself   1  None   10     default  1   3000ms
e1/1/12  12.12.12.1      SM   Dis  Itself   1  None   None   default  1   3000ms
v20      21.21.21.22     SM   Ena  Itself   1  None   None   default  1   3000ms
v60      60.60.60.1      SM   Ena  Itself   1  None   None   default  1   3000ms
v310     110.110.110.2   SM   Dis  Itself   1  None   None   default  1   3000ms
v360     160.160.160.1   SM   Dis  Itself   1  None   None   default  1   3000ms
l2       4.4.4.4         SM   Ena  Itself   1  None   None   default  1   3000ms
l3       10.10.10.10     SM   Ena  Itself   1  None   None   default  1   3000ms
Total Number of Interfaces : 9
```

## History

| Release version | Command history |
|---|---|
| 8.0.20a | This command was modified to display neighbor filter information. |

# show ip pim mcache

Displays the PIM multicast cache.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **mcache** [ *source-address* | *group-address* | **counts** | **dense** | [ **dit-idx** *dit-idx* | **g_entries** | **receiver** | **sg_entries** | **sparse** | **ssm** ]

## Parameters

**vrf** *vrf-name*
Specifies information for a VRF instance.

*source-address*
Specifies the multicast cache source address.

*group-address*
Specifies the multicast cache group address.

**counts**
Specifies the number of entries.

**dense**
Specifies displaying only the PIM Dense Mode entries.

**dit-idx** *dit-idx*
Specifies displaying all entries that match a specified downstream interface (DIT).

**g_entries**
Specifies displaying only the (*, G) entries.

**receiver**
Specifies displaying all entries that egress a specified interface.

**sg_entries**
Specifies displaying only the (S, G) entries.

**sparse**
Specifies displaying only the PIM Sparse Mode entries.

**ssm**
Specifies displaying only the SSM entries.

## Modes

Privileged EXEC mode

## Command Output

The **show ip pim mcache** command displays the following information:

| Output Field | Description |
|---|---|
| Total entries in mcache | The total number of PIM mcache entries |
| MJ | Membership Join |
| MI | Membership Include |
| ME | Membership Exclude - Legend for the mcache entry printed once per page, it gives the explanation of each of the flags used in the entry. |
| BR | Blocked RPT |
| BA | Blocked Assert |
| BF | Blocked Filter |
| BI | Blocked IIF |
| Uptime | Shows the entry uptime |
| upstream neighbor | Shows the upstream neighbor for the Source/RP based on the type of entry. For (*,G) it shows the upstream neighbor towards the RP. For (S,G) entries it shows the upstream neighbor towards the source. |
| Flags | Flags Represent Entry flags in hex format in the braces. And indicates the meaning of the flags set in abbreviated string whose explanations are as below. Only shows the flags which are set.<br>SM - Shows If the entry is created by PIM Sparse Mode<br>DM - Shows If DM mode entry is enabled<br>SSM - Shows If the SSM mode entry is enabled<br>RPT - Shows If the entry is on the rendezvous point (RP)<br>SPT - Shows If the entry is on the source tree<br>LSRC - Shows If the source is in a directly-connected interface<br>LRcv - Shows If the receiver is directly connected to the router<br>REG - if the data registration is in progress<br>L2REG - if the source is directly connected to the router<br>REGSUPP - if the register suppression timer is running<br>RegProbe<br>HW - Shows If the candidate for hardware forwarding is enabled<br>FAST - Shows If the resources are allocated for hardware forwarding<br>TAG - Shows If there is a need for allocating entries from the replication table<br>MSDPADV - Shows If RP is responsible for the source and must be advertised to its peers.<br>NEEDRTE - Shows If there is no route to the source and RP is available<br>PRUNE - Shows If PIM DM Prune to upstream is required |
| RP | Shows the IP address of the RP. |
| fast ports | Shows forwarding port mask. |
| AgeSltMsk | Shows a value of 1 if the entry is programmed in hardware, and a value of 0 if it is not programmed in hardware. |

| Output Field | Description |
|---|---|
| L2 FID | Shows the hardware resource allocated for the traffic switched to receivers in the ingress VLAN. |
| DIT | Shows the hardware resource allocated for routed receivers. |
| RegPkt | Shows Count of Packets forwarded due to the Register decapsulation. |
| Number of matching entries | Shows the total number of mcache entries matching a particular multicast filter specified. |
| Outgoing interfaces Section | This section consists of three parts. L3 OIFs, L2OIFs and Blocked OIFs. And each section has Format of L3/L2/Blocked followed by (HW/SW) followed by count of the number of OIF in each section.<br>Additionally, each section displays the OIFs one per line. And shows the OIF in the format eth/Tr(Vlan) followed by uptime/ expiry time, followed by the Flags associated with each OIF. |
| L3 | Shows whether the traffic is routed out of the interface. |
| L2 | Shows whether the traffic is switched out of the interface. |
| HW | Shows whether the entry is hardware forwarded. |
| SW | Shows whether the entry is software forwarded |
| Eth/Tr(VL1) | Shows the outgoing interface on the specified VLAN. |
| Flags (explanation of flags in the OIF section) | Shows the flags set in each of the Outgoing interface in abbreviated string format whose explanations are as below. Legend of this shown at the top of each entry<br>IM - Immediate<br>IH - Inherited<br>MJ - Membership Join<br>MI - Membership Include<br>ME - Membership Exclude<br>BR - Blocked due to SG RPT<br>BA - Blocked due to Assert<br>BF - Blocked due to Filter<br>BI - Blocked IIF (Incoming interface) matches OIF |
| Src-Vlan | Shows the VLAN associated with the ingress interface. |
| MCTPEERF - Traffic Forw By Cluster Peer CCEP | Applies only to Layer 3 multicast routing over MCT. This means multicast traffic for this stream is forwarded by cluster peer [remote] CCEP port because of flow load balancing |

# Examples

This example shows all PIM multicast cache entries:

```
Device(config)# show ip pim mcache
IP Multicast Mcache Table
Entry Flags : SM - Sparse Mode, SSM - Source Specific Multicast, DM - Dense Mode
       RPT - RPT Bit, SPT - SPT Bit, LSRC - Local Source, LRCV - Local Receiver
       HW - HW Forwarding Enabled, FAST - Resource Allocated, TAG - Need For
        Replication Entry
       REGPROB - Register In Progress, REGSUPP - Register Suppression Timer
       MSDPADV - Advertise MSDP, NEEDRTE - Route Required for Src/RP, PRUN - DM
        Prune Upstream
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
       MJ - Membership Join, MI - Membership Include, ME - Membership Exclude
       BR - Blocked RPT, BA - Blocked Assert, BF - Blocked Filter, BI - Blocked IIF
Total entries in mcache: 20
1  (140.140.140.3, 225.0.0.1) in v340 (tag e1/8/1), Uptime 00:00:02
  Source is directly connected
  Flags (0x200004e1) DM HW FAST TAG
  fast ports: ethe 1/4/6 ethe 1/8/26
  AgeSltMsk: 1, L2 FID: 8188, DIT: 3
  Forwarding_oif: 2
  L3 (HW) 2:
    TR(e1/4/6,e1/4/6)(VL330), 00:00:02/0, Flags: IM
    e1/8/26(VL310), 00:00:02/0, Flags: IM
  Src-Vlan: 340
```

This example shows the PIM multicast cache for the specified address:

```
Device(config)# show ip pim mcache 10.140.140.14 230.1.1.9
IP Multicast Mcache Table
Entry Flags : SM - Sparse Mode, SSM - Source Specific Multicast, DM - Dense Mode
     RPT - RPT Bit, SPT - SPT Bit, LSRC - Local Source, LRCV - Local Receiver
     HW - HW Forwarding Enabled, FAST - Resource Allocated, TAG - Need For Replication Entry
     REGPROB - Register In Progress, REGSUPP - Register Suppression Timer
     MSDPADV - Advertise MSDP, NEEDRTE - Route Required for Src/RP, PRUN - DM Prune Upstream
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
     MJ - Membership Join, MI - Membership Include, ME - Membership Exclude
     BR - Blocked RPT, BA - Blocked Assert, BF - Blocked Filter, BI - Blocked IIF
Total entries in mcache: 20
1 (10.140.140.14, 230.1.1.9) in v1001 (tag e1/4/29), Uptime 00:03:12
  upstream neighbor 10.11.11.13
  Flags (0x600680e1) SM SPT LRCV HW FAST TAG
  fast ports: ethe 1/4/29 ethe 1/5/2
  AgeSltMsk: 1, L2 FID: 8188, DIT: 8
  Forwarding_oif: 3, Immediate_oif: 0, Blocked_oif: 0
  L3 (HW) 2:
  e1/4/29(VL13), 00:03:12/0, Flags: MJ
  e1/5/2(VL1004), 00:03:12/0, Flags: MJ
  L2 (HW) 1:
  e1/5/2, 00:00:07/0, Flags: MJ
  L2 MASK: ethe 1/5/2
  Src-Vlan: 1001
```

This example shows the PIM multicast cache for the specified DIT:

```
Device# show ip pim mcache dit-idx 2
IP Multicast Mcache Table
Entry Flags    : SM  - Sparse Mode, SSM - Source Specific Multicast, DM - Dense Mode
                 RPT    - RPT Bit, SPT - SPT Bit, LSRC - Local Source, LRCV - Local Receiver
                 HW - HW Forwarding Enabled, FAST - Resource Allocated, TAG - Need For Replication
Entry
                 REGPROB - Register In Progress, REGSUPP - Register Suppression Timer
                 MSDPADV - Advertise MSDP, NEEDRTE - Route Required for Src/RP,  PRUN - DM Prune
Upstream
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
                 MJ - Membership Join, MI - Membership Include, ME - Membership Exclude
                 BR - Blocked RPT, BA - Blocked Assert, BF - Blocked Filter, BI - Blocked IIF
Total entries in mcache: 30
1     (20.20.20.100, 225.1.1.1) in v220 (tag e1/1/13), Uptime 07:12:07
      upstream neighbor 220.220.220.1
      Flags (0x200680e1) SM SPT LRCV HW FAST TAG
      fast ports: ethe 1/1/11
      AgeSltMsk: 1,  L2 FID: 105c,  DIT:     2
      Forwarding_oif: 1, Immediate_oif: 0, Blocked_oif: 0
      L3 (HW) 1:
          e1/1/11(VL40), 07:12:07/0, Flags: MJ
      Src-Vlan: 220
2     (20.20.20.100, 225.1.1.2) in v220 (tag e1/1/13), Uptime 00:01:00
      upstream neighbor 220.220.220.1
      Flags (0x200680e1) SM SPT LRCV HW FAST TAG
      fast ports: ethe 1/1/11
      AgeSltMsk: 1, L2 FID: 105c,  DIT:     2
      Forwarding_oif: 1, Immediate_oif: 0, Blocked_oif: 0
      L3 (HW) 1:
          e1/1/11(VL40), 00:01:00/0, Flags: MJ
      Src-Vlan: 220
3     (20.20.20.100, 225.1.1.3) in v220 (tag e1/1/13), Uptime 00:01:00
      upstream neighbor 220.220.220.1
      Flags (0x200680e1) SM SPT LRCV HW FAST TAG
      fast ports: ethe 1/1/11
      AgeSltMsk: 1, L2 FID: 105c,  DIT:     2
      Forwarding_oif: 1, Immediate_oif: 0, Blocked_oif: 0
      L3 (HW) 1:
          e1/1/11(VL40), 00:01:00/0, Flags: MJ
      Src-Vlan: 220
```

This example shows the PIM multicast cache with Layer 3 multicast routing over MCT, showing that multicast traffic for a stream is forwarded by a cluster peer CCEP port because of flow load balancing.

```
Device# show ip pim mcache
IP Multicast Mcache Table
Entry Flags    : SM  - Sparse Mode, SSM - Source Specific Multicast, DM - Dense Mode
                 RPT    - RPT Bit, SPT - SPT Bit, LSRC - Local Source, LRCV - Local Receiver
                 HW - HW Forwarding Enabled, FAST - Resource Allocated, TAG - Need For Replication Entry
                 REGPROB - Register In Progress, REGSUPP - Register Suppression Timer
                 MSDPADV - Advertise MSDP, NEEDRTE - Route Required for Src/RP,  PRUN - DM Prune
Upstream
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert, MCTPEERF - Traffic Forw By Cluster
Peer CCEP
                 MJ - Membership Join, MI - Membership Include, ME - Membership Exclude
                 BR - Blocked RPT, BA - Blocked Assert, BF - Blocked Filter, BI - Blocked IIF
Total entries in mcache: 2

1     (39.39.39.1, 229.1.1.10) in v40 (tag e2/1/12), Uptime 00:21:31
      upstream neighbor 40.40.40.175
      Flags (0x200284e1) SM SPT HW FAST TAG
      fast ports: ethe 2/1/11
      AgeSltMsk: 1, IPMC:     4
      Forwarding_oif: 1, Immediate_oif: 1, Blocked_oif: 0
      L3 (HW) 1:
          TR(e2/1/11,e2/1/11)(VL10), 00:21:31/178, Flags: IM  MCTPEERF
      Src-Vlan:   40
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | The output of the command was modified to remove the AvgRate and Profile entries. |
| 8.0.30h | The output of the command was modified to remove the rate counter. |
| 8.0.40a | The output of the command was modified to remove the rate counter. |
| 8.0.30 | This command was modified to show output for Layer 3 multicast routing over MCT. |

# show ip pim neighbor

Displays information about PIM neighbors.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **neighbor** [ **ethernet** *stack/slot/port* | **tunnel** *tunnel-id* | **ve** *ve-num* ]

## Parameters

**vrf** *vrf-name*
Displays information for the specified VRF instance.

**ethernet** *stack/slot/port*
Displays information for the specified Ethernet interface.

**tunnel** *tunnel-id*
Displays information for the specified Tunnel interface.

**ve** *ve-num*
Displays information for the specified VE interface.

## Modes

User EXEC mode

## Command Output

The **show ip pim neighbor** command displays the following information:

| Output Field | Description |
|---|---|
| Port | The interface through which the device is connected to the neighbor. |
| Phyport | When there is a virtual interface, this is the physical port to which the neighbor is connected. |
| Neighbor | The IP interface of the PIM neighbor. |
| Holdtime sec | Indicates how many seconds the neighbor wants this device to hold the entry for this neighbor in memory. The neighbor sends the Hold Time in Hello packets:<br>• If the device receives a new Hello packet before the Hold Time received in the previous packet expires, the device updates its table entry for the neighbor.<br>• If the device does not receive a new Hello packet from the neighbor before the Hold time expires, the device assumes the neighbor is no longer available and removes the entry for the neighbor. |
| Age sec | The number of seconds since the device received the last hello message from the neighbor. |
| UpTime sec | The number of seconds the PIM neighbor has been up. This timer starts when the device receives the first Hello messages from the neighbor. |

| Output Field | Description |
|---|---|
| VRF | The VRF in which the interface is configured. This can be a VRF that the port was assigned to or the default VRF of the device. |
| Priority | The DR priority that is used in the DR election process. This can be a configured value or the default value of 1. |

# Examples

The following example shows information about PIM neighbors.

```
device(config)# show ip pim neighbor
--------+--------+---------------+--------+---+---------+---------+-----+------------+-----------+----
Port    |PhyPort |Neighbor       |Holdtime|T  |PropDelay|Override |Age  |UpTime      |VRF        |Prio
        |        |               |sec     |Bit|msec     |msec     |sec  |            |           |
--------+--------+---------------+--------+---+---------+---------+-----+------------+-----------+----
v2       e1/1/1     2.1.1.2         105      1   500       3000      0     00:44:10  default-vrf 1
v4       e1/2/2     4.1.1.2         105      1   500       3000      10    00:42:50  default-vrf 1
v5       e1/1/4     5.1.1.2         105      1   500       3000      0     00:44:00  default-vrf 1
v22      e1/1/1     22.1.1.1        105      1   500       3000      0     00:44:10  default-vrf 1
Total Number of Neighbors : 4
```

# show ip pim nsr

Displays the multicast nonstop routing (NSR) status information.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **nsr**

## Parameters

**vrf** *vrf-name*
Specifies information for a VRF instance.

## Modes

User EXEC mode

## Usage Guidelines

When entered without the **vrf** option, this command displays information for the default VRF instance.

## Command Output

The **show ip pim nsr** command displays the following information:

| Output Field | Description |
|---|---|
| NSR | The NSR field indicates whether the **ip multicast-nonstop-routing** command is enabled (ON) or disabled (OFF). |
| Switchover in Progress Mode | The Switchover in Progress Mode field indicates whether the multicast traffic is in the middle of a switchover (displaying a TRUE status), or not (displaying a FALSE status). |

## Examples

The following example displays PIM NSR status information.

```
device> show ip pim nsr
Global Mcast NSR Status
  NSR: ON
  Switchover In Progress Mode: FALSE
```

# show ip pim optimization

Displays PIM optimization information.

## Syntax

**show ip pim optimization** [ **dit-idx** | **vlan-fid** ]

## Parameters

**dit-idx**

Represents the IPMC index.

**vlan-fid**

Represents the software VLAN index that stores the Layer 2 OIF sets.

## Modes

User EXEC mode

Privileged EXEC mode

## Command Output

The **show ip pim optimization** command displays the following information:

| Output field | Description |
|---|---|
| IPMC | The IP multicast entry number. |
| SetId | Identifies the internal software resource used in sharing (optimizing). |
| Set | The set manager database ID. |
| SW-VIDX | The internal software VLAN index used for sharing Layer 2 OIFs. |

## Examples

The following example displays optimization information for all VRFs.

```
device# show ip pim optimization dit-idx
Displaying Optimization information for all vrfs
Total IPMCs Allocated:    2; Available: 7831; Failed:    0
Index    IPMC          SetId         Users         Set
   1.      374         0x30eb38c8      100 {[VLAN <30>:Port <37/2/2>],}
   2.      363         0x305dd728      100 {[VLAN <30>:Port
<36/1/48>],[VLAN <30>:Port <31/1/48>],
                                             [VLAN <30>:Port <24/1/48>],}
Sharability Coefficient:  99%
```

The following example displays the PIM optimization vlan-fid information.

```
device# show ip pim optimization vlan-fid
Total SW-VIDXs Allocated:    2; Available: 4093; Failed:    0
Index    SW-VIDX       SetId         Users         Set
   1.       1          0x30e98448        1 {Port <37/2/2>,}
   2.       5          0x305d68a0        1 {Port <36/1/48>,Port
<31/1/48>,Port <24/1/48>,}
Sharability Coefficient:   0%
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ip pim prune

Displays all multicast cache entries that are currently in a pruned state and have not yet aged out.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **prune**

## Parameters

**vrf** *vrf-name*

Displays information for a specific VRF instance.

## Modes

Privileged EXEC mode

## Usage Guidelines

When entered without the **vrf** option, this command displays information for the default VRF instance.

## Examples

This example shows all multicast cache entries that are currently in a pruned state and have not yet aged out:

```
device> show ip pim prune
    1 (104.1.1.2 231.0.1.1):
   e1/2/2,1/2/2(150)
    2 (108.1.1.100 231.0.1.1):
   e1/2/2,1/2/2(150)
    3 (104.1.1.2 231.0.1.2):
   e1/2/2,1/2/2(150)
    4 (108.1.1.100 231.0.1.2):
   e1/2/2,1/2/2(150)
    5 (108.1.1.100 231.0.1.3):
   e1/2/2,1/2/2(150)
    6 (104.1.1.2 231.0.1.4):
   e1/2/2,1/2/2(150)
    7 (108.1.1.100 231.0.1.4):
   e1/2/2,1/2/2(150)
    8 (104.1.1.2 231.0.1.5):
   e1/2/2,1/2/2(150)
    9 (108.1.1.100 231.0.1.5):
   e1/2/2,1/2/2(150)
Total Prune entries: 9
```

# show ip pim resource

Displays the hardware resource information, such as hardware allocation, availability, and limit, for software data structures.

## Syntax

**show ip pim** [ **all-vrf** | **vrf** *vrf-name* ] **resource**

## Parameters

**all-vrf**

Displays information for all virtual routing and forwarding instances (VRFs).

**vrf** *vrf-name*

Displays information for a particular VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ip pim resource** command displays the following information:

| Output Field | Description |
|---|---|
| Num alloc | Number of VRF instances allocated. |
| System max | Maximum number of VRFs allowed in the system. |
| Size | Size of one instance of the resource in bytes. |
| alloc | Number of nodes of that data that are currently allocated in memory. |
| in-use | Number of allocated nodes in use. |
| avail | Number of allocated nodes are not in use. |
| get-fail | Number of allocation failures for this node. |
| limit | Maximum number of nodes that can be allocated for a data structure. This may or may not be configurable, depending on the data structure |
| get-mem | Number of successful allocations for this node. |
| size | Size of the node in bytes. |
| init | Number of nodes that are allocated during initialization time. |

# Examples

The following example displays output from the **show ip pim resource** command.

```
device> show ip pim resource
Global PIM Parameters :-
GLOBAL Ipv4 MULTICAST CLASS Size:16811 bytes
GLOBAL Ipv4 PIM CLASS Size:1065 bytes
MULTICAST IPV4 CLASS Num alloc:5, System max:129, Size:1228 bytes
PIM IPV4 CLASS  Num alloc:5, System max:129, Size:50440
 Vrf Instance : default-vrf
-------------------------------------
                     alloc in-use  avail get-fail    limit  get-mem  size init
NBR list               256      3    253        0      512        4    90  256
RP set list            256      4    252        0     1536     5032    43  256
Static RP               64      0     64        0       64        0    36   64
LIF Entry              512      0    512        0      512        0    41  512
Anycast RP              64      0     64        0       64        0   190   64
timer                  256      0    256        0    59392        4    64  256
prune                  128      0    128        0    29696        0    34  128
pimsm J/P elem        1024      0   1024        0    48960     1258    29 1024
Timer Data             256      1    255        0    59392        2    28  256
mcache SLIB Sync       280      0    280        0    64960       20    28  280
mcache                  56      2     54        0    12992        2   796   56
graft if no mcache     197      0    197        0    45704        0    64  197
HW replic vlan        2000      3   1997        0   464000        4    66 2000
HW replic port        1024      3   1021        0   237568        4    78 1024
pim/dvm intf. group    256      0    256        0    59392        0    24  256
pim/dvm global group   256      2    254        0    59392        2    46  256
repl entry(Global)    1024      0   1024        0   237568        4    43 1024
IGMP Resources(All Vrfs):
  groups               256      2    254        0     4096        2   210  256
  group-memberships    256      2    254        0     4096        2   142  256
  sources               56      1     55        0    12992      606    59   56
  client sources        56      0     56        0    12992        0    81   56
  ssm-map              256      0    256        0      256        0    18  256
  ssm-map-sources      256      0    256        0    59392        0  1024  256
Hardware-related Resources:
Total (S,G) entries 1
Total SW FWD entries 0
  Total sw w/Tag MVID entries 0
  Total sw w/Tag invalid MVID entries 0
Total HW FWD entries 1
  Total hw w/Tag MVID entries 0
  Total hw w/Tag invalid MVID entries 0
```

# show ip pim rp-candidate

Displays candidate rendezvous point (RP) information.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **rp-candidate**

## Parameters

**vrf** *vrf-name*

Displays information for a specific VRF instance.

## Modes

User EXEC mode

## Usage Guidelines

When used without the **vrf** option, this command displays information for the default VRF.

## Command Output

The **show ip pim rp-candidate** command displays the following information:

| Output Field | Description |
|---|---|
| Candidate-RP-advertisement in | How time will pass before the BSR sends the next RP message. The time is displayed in "hh:mm:ss" format.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |
| RP | The IP address of the RP.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |
| group prefixes | The multicast groups for which the RP listed by the previous field is a candidate RP.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |
| Candidate-RP-advertisement period | How frequently the BSR sends candidate RP advertisement messages.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |

# Examples

The following example shows information for a candidate RP.

```
device> show ip pim rp-candidate
Next Candidate-RP-advertisement in 00:00:10
  RP: 207.95.7.1
    group prefixes:
    224.0.0.0 / 4
  Candidate-RP-advertisement period: 60
```

# show ip pim rpf

Displays what PIM sees as the best reverse path to the source. While there may be multiple routes back to the source, the one displayed by this command is the one that PIM thinks is best.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **rpf** *ip-address* [ *group-address* ]

## Parameters

**vrf** *vrf-name*
Displays information for the specified VRF instance.

*ip-address*
Specifies the source address for reverse-path forwarding (RPF) check.

*group-address*
Specifies the group address for reverse-path forwarding (RPF) check.

## Modes

User EXEC mode

## Examples

This example shows best reverse path to the specified source:

```
device# show ip pim vrf eng rpf 130.50.11.10
Source 130.50.11.10 directly connected on e1/4/1
```

# show ip pim rp-hash

Displays rendezvous-point (RP) information for a PIM Sparse group.

## Syntax

**show ip pim** [**vrf** *vrf-name* ] **rp-hash** *group-addr*

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

Privileged EXEC mode

## Command Output

The **show ip pim rp-hash** command displays the following information:

| Output Field | Description |
|---|---|
| RP | Indicates the IP address of the RP for the specified PIM Sparse group. |
| Info source | Indicates the source of the RP information. It can be a static-RP configuration or learned via the bootstrap router. If RP information is learned from the boot strap, the BSR IP address is also displayed. |

## Examples

The following example shows RP information for a PIM Sparse group.

```
device# show ip pim rp-hash 239.255.162.1
  RP: 207.95.7.1, v2
    Info source: 207.95.7.1, via bootstrap
```

# show ip pim rp-map

Displays rendezvous-point (RP)-to-group mapping information.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **rp-map**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ip pim rp-map** command displays the following information:

| Output Field | Description |
|---|---|
| Group address | Indicates the PIM Sparse multicast group address using the listed RP. |
| RP address | Indicates the IP address of the RP for the listed PIM Sparse group. |

## Examples

The following example shows RP-to-group mapping.

```
device> show ip pim rp-map
Number of group-to-RP mappings: 6
Group address        RP address
------------------------------
1 239.255.163.1  99.99.99.5
2 239.255.163.2  99.99.99.5
3 239.255.163.3  99.99.99.5
4 239.255.162.1  99.99.99.5
5 239.255.162.2  43.43.43.1
6 239.255.162.3  99.99.99.5
```

# show ip pim rp-set

Displays rendezvous-point (RP)-set list for the device elected as the bootstrap router (BSR).

## Syntax

**show ip pim** [ **all-vrf** | **vrf** *vrf-name* ] **rp-set**

## Parameters

**all-vrf**

Displays information for all VRF instances.

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ip pim rp-set** command displays the following information:

| Output Field | Description |
|---|---|
| Number of group prefixes | The number of PIM Sparse group prefixes for which the RP is responsible. |
| Group prefix | Indicates the multicast groups for which the RP listed by the previous field is a candidate RP. |
| RPs expected or received | Indicates how many RPs were expected and received in the latest bootstrap message. |
| RP *num* | Indicates the RP number. If there are multiple RPs in the PIM Sparse domain, a line of information for each RP is listed, in ascending numerical order. |
| priority | The RP priority of the candidate RP. During the election process, the candidate RP with the highest priority is elected as the RP. |
| age | The age (in seconds) of this RP-set. |
| holdtime | Indicates the time in seconds for which this rp-set information is valid.<br>If this rp-set information is not received from BSR within the holdtime period, the rp-set information is aged out and deleted. |

# Examples

The following example shows the RP set list for the device elected as BSR.

```
device> show ip pim rp-set
Static RP
---------
Static RP count: 2
1.51.51.4
1.51.51.5
Number of group prefixes Learnt from BSR: 1
Group prefix = 224.0.0.0/4      # RPs: 2
      RP 1: 1.51.51.1    priority=0    age=60    holdtime=150
      RP 2: 1.51.51.3    priority=0    age=30    holdtime=150
```

The following example shows the RP set list for devices that are not elected as BSR.

```
device> show ip pim rp-set
Static RP
---------
Static RP count: 2
1.51.51.4
1.51.51.5
Number of group prefixes Learnt from BSR: 1
Group prefix = 224.0.0.0/4      # RPs expected: 2
      # RPs received: 2
      RP 1: 1.51.51.1    priority=0    age=60    holdtime=150
      RP 2: 1.51.51.3    priority=0    age=30    holdtime=150
```

# show ip pim sparse

Displays PIM Sparse configuration information, including whether the hardware-drop feature is enabled or disabled, and information for PIM SSM range ACL configuration.

## Syntax

**show ip pim** [ **vrf** *vrf-name* ] **sparse**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ip pim sparse** command displays the following information:

| Output field | Description |
|---|---|
| **Global PIM Sparse mode settings** | |
| Maximum mcache | Maximum number of multicast cache entries. |
| Current Count | Number of multicast cache entries used. |
| Hello interval | How often the device sends IPIM Sparse hello messages to its PIM Sparse neighbors. This field shows the number of seconds between hello messages. PIM Sparse routers use hello messages to discover one another. |
| Neighbor timeout | Number of seconds the device waits for a hello message from a neighbor before determining that the neighbor is no longer present and is not removing cached PIM Sparse forwarding entries for the neighbor. The default is 105 seconds. |
| Join or Prune interval | How frequently the device sends IPv6 PIM Sparse Join or Prune messages for the multicast groups it is forwarding. This field shows the number of seconds between Join or Prune messages.<br>The device sends Join or Prune messages on behalf of multicast receivers that want to join or leave an PIM Sparse group. When forwarding packets from PIM Sparse sources, the device sends the packets only on the interfaces on which it has received join requests in Join or Prune messages for the source group. |
| Inactivity interval | Number of seconds a forwarding entry can remain unused before the router deletes it. The default is 180 seconds. |
| Hardware Drop Enabled | Whether hardware drop is enabled or disabled.<br>To prevent unwanted multicast traffic from being sent to the CPU, PIM Routing and Passive Multicast Route Insertion (PMRI) can be used together to ensure that multicast streams are only |

| Output field | Description |
|---|---|
| | forwarded out ports with interested receivers and unwanted traffic is dropped in the hardware on Layer 3 Switches. |
| Prune Wait Interval | Number of seconds a PIM device waits before stopping traffic to neighbor devices that do not want the traffic. The range is 0 to 3 seconds. The default is 3 seconds. |
| Bootstrap Msg interval | How frequently the BSR configured on the device sends the RP set to the RPs within the PIM Sparse domain. The RP set is a list of candidate RPs and their group prefixes. The group prefix of a candidate RP indicates the range of PIM Sparse group numbers for which it can be an RP.<br><br>**NOTE**<br>This field contains a value only if an interface on the device is elected to be the BSR. Otherwise, the field is blank. |
| Candidate-RP Msg interval | Number of seconds the candidate RP configured on the Layer 3 switch sends candidate RP advertisement messages to the BSR. The default is 60 seconds. |
| Register Suppress Time | This is the mean interval between receiving a Register-Stop and allowing registers to be sent again. A lower value means more frequent register bursts at RP, while a higher value means longer join latency for new receivers. The default is 60 seconds. |
| Register Probe Time | Number of seconds the PIM router waits for a register-stop from an RP before it generates another NULL register to the PIM RP. The default is 10 seconds. |
| Register Stop Delay | Register stop message. The default is 10 seconds. |
| Register Suppress interval | Number of seconds that it takes the designated router to send a Register-encapsulated date to the RP after receiving a Register-Stop message. The default is 60 seconds. |
| SSM Enabled | If yes, source-specific multicast is configured globally on this router. |
| SPT threshold | Number of packets the device sends using the path through the RP before switching to the SPT path. The default is 1 packet. |
| SSM Group Range | Source-specific multicast group range. |
| Route Precedence | The route precedence configured to control the selection of routes based on the four route types:<br>• Non-default route from the mRTM<br>• Default route from the mRTM<br>• Non-default route from the uRTM<br>• Default route from the uRTM |

# Examples

The following example displays PIM Sparse configuration information.

```
device> show ip pim sparse
Global PIM Sparse Mode Settings
  Maximum Mcache         : 12992       Current Count              : 0
  Hello interval         : 30          Neighbor timeout           : 105
  Join/Prune interval    : 60          Inactivity interval        : 180
  Hardware Drop Enabled  : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time : 60          Register Probe Time        : 10
  Register Stop Delay    : 10          Register Suppress interval : 60
  SSM Enabled            : Yes         SPT Threshold              : 1
  SSM Group Range        : 232.0.0.0/8
  Route Precedence       : mc-non-default mc-default uc-non-default uc-default
```

The following example displays PIM Sparse configuration for a VRF instance named my_vrf.

```
device> show ip pim my_vrf sparse
Global PIM Sparse Mode Settings
  Maximum Mcache         : 12992       Current Count              : 0
  Hello interval         : 30          Neighbor timeout           : 105
  Join/Prune interval    : 60          Inactivity interval        : 180
  Hardware Drop Enabled  : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time : 60          Register Probe Time        : 10
  Register Stop Delay    : 10          Register Suppress interval : 60
  SSM Enabled            : Yes         SPT Threshold              : 1
  SSM Group Range        : 232.0.0.0/8
  Route Precedence       : mc-non-default mc-default uc-non-default uc-default
```

This example shows whether the hardware-drop feature has been enabled or disabled:

```
device> show ip pim sparse
Global PIM Sparse Mode Settings
  Maximum Mcache         : 12992       Current Count              : 0
  Hello interval         : 30          Neighbor timeout           : 105
  Join/Prune interval    : 60          Inactivity interval        : 180
  Hardware Drop Enabled  : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time : 60          Register Probe Time        : 10
  Register Stop Delay    : 10          Register Suppress interval : 60
  SSM Enabled            : Yes         SPT Threshold              : 1
  SSM Group Range        : 232.0.0.0/8
  Route Precedence       : mc-non-default mc-default uc-non-default uc-default
```

The following example displays information for PIM SSM range ACL configuration.

```
device> show ip pim sparse
Global PIM Sparse Mode Settings
  Maximum Mcache         : 0           Current Count              : 0
  Hello interval         : 30          Neighbor timeout           : 105
  Join/Prune interval    : 60          Inactivity interval        : 180
  Register Suppress Time : 60          Register Probe Time        : 10
  SPT Threshold          : 1           Hardware Drop Enabled      : Yes
  Bootstrap Msg interval : 60          Candidate-RP Msg interval  : 60
  Register Stop Delay    : 60          Register Suppress interval : 60
  SSM Enabled            : Yes
  SSM Group Range        : 224.1.1.1/24
  SSM Group Range ACL    : xyz
  Route Precedence       : mc-non-default mc-default uc-non-default uc-default
```

# show ip pim traffic

Displays IPv4 PIM traffic statistics.

## Syntax

**show ip pim traffic** [ **vrf** *vrf-name* ] [ **join-prune** ] [ **rx** | **tx** ]

## Parameters

**vrf** *vrf-name*

Specifies information for a VRF instance.

**join-prune**

Specifies displaying join and prune statistics.

**rx**

Specifies displaying received PIM traffic statistics.

**tx**

Specifies displaying transmitted PIM traffic statistics.

## Modes

Privileged EXEC mode

## Usage Guidelines

PIM control packet statistics for interfaces that are configured for standard PIM are listed first by the display.

## Command Output

The **show ip pim traffic** command displays the following information:

| Output Field | Description |
|---|---|
| Port | The port or virtual interface on which the PIM interface is configured. |
| HELLO | The number of PIM Hello messages sent or received on the interface. |
| JOIN-PRUNE | The number of Join or Prune messages sent or received on the interface.<br><br>**NOTE**<br>Unlike PIM Dense, PIM Sparse uses the same messages for Joins and Prunes. |
| ASSERT | The number of Assert messages sent or received on the interface. |
| REGISTER GRAFT (DM) | The number of Register messages sent or received on the interface. |

| Output Field | Description |
|---|---|
| REGISTER STOP (SM) | The number of Register Stop messages sent or received on the interface. |
| BOOTSTRAP MSGS (SM) | The number of bootstrap messages sent or received on the interface. |
| CAND. RP ADV. (SM) | The total number of Candidate-RP-Advertisement messages sent or received on the interface. |
| Err | The total number of messages discarded, including a separate counter for those that failed the checksum comparison. |

# Examples

This example shows PIM join and prune traffic statistics for received and sent packets:

```
device(config)# show ip pim traffic
Port    HELLO     JOIN-PRUNE  ASSERT    REGISTER  REGISTER  BOOTSTRAP CAND. RP  Err
                                        GRAFT(DM) STOP(SM)  MSGS (SM) ADV. (SM)
------+---------+-----------+---------+---------+---------+---------+---------+---
       Rx        Rx          Rx         Rx        Rx        Rx        Rx        Rx
------+---------+-----------+---------+---------+---------+---------+---------+---
v30     0         0           0         0         0         0         0         0
v50     2526      1260        0         0         0         1263      0         0
v150    2531      0           0         0         0         1263      0         0
v200    2531      0           0         0         0         1         0         0
Port    HELLO     JOIN-PRUNE  ASSERT    REGISTER  REGISTER  BOOTSTRAP CAND. RP  Err
                                        GRAFT(DM) STOP(SM)  MSGS (SM) ADV. (SM)
------+---------+-----------+---------+---------+---------+---------+---------+---
       Tx        Tx          Tx         Tx        Tx        Tx        Tx
------+---------+-----------+---------+---------+---------+---------+---------+---
v30     2528      0           0         0         0         0         0
v50     2540      1263        0         0         0         2         0
v150    2529      0           0         0         0         1262      0
v200    2529      0           0         0         0         1262      0
```

This example shows the number of received IPv4 PIM Hello packets dropped on interface 1/1/9 because an ACL to control neighbor access is configured on it.

```
device# show ip pim traffic rx
Port    HLO   JN-PRNE  ASSERT REG       REG        BTSTRP    CAND RP   Err
                        GRFT(DM) STOP(SM)   MSGS(SM)   ADV.(SM)

-------+-------+------+-----+-------+----------+---------+---------+---
        Rx      Rx     Rx    Rx      Rx         Rx        Rx        Rx
-------+-------+------+-----+-------+----------+---------+---------+---
e1/1/1  0       0      0     0       0          0         0         0
e1/1/9  764     0      0     0       0          0         0         757
e1/1/12 0       0      0     0       0          0         0         0
v20     758     0      0     1916    0          0         0         0
v60     0       0      0     0       0          0         0         0
v310    0       0      0     0       0          0         0         0
v360    0       0      0     0       0          0         0         0
```

This example shows PIM join and prune traffic statistics for sent packets:

```
device(config)# show ip pim traffic tx
Port    HELLO     JOIN-PRUNE  ASSERT    REGISTER  REGISTER  BOOTSTRAP CAND. RP  Err
                                        GRAFT(DM) STOP(SM)  MSGS (SM) ADV. (SM)
------+---------+-----------+---------+---------+---------+---------+---------+---
       Tx        Tx          Tx         Tx        Tx        Tx        Tx
------+---------+-----------+---------+---------+---------+---------+---------+---
v30     2528      0           0         0         0         0         0
v50     2540      1263        0         0         0         2         0
v150    2529      0           0         0         0         1262      0
v200    2530      0           0         0         0         1262      0
```

This example shows PIM join and prune traffic statistics.

```
device(config)# show ip pim traffic join-prune
Port  Packet    Join      Prune     Avg Aggr  Last Aggr
-----+---------+---------+---------+---------+----------
      Rx        Rx        Rx        Rx        Rx
-----+---------+---------+---------+---------+----------
v30   0         0         0         0         0
v50   1260      1260      0         1         1
v150  0         0         0         0         0
v200  0         0         0         0         0
Port  Packet    Join      Prune     Avg Aggr  Last Aggr
-----+---------+---------+---------+---------+----------
      Tx        Tx        Tx        Tx        Tx
-----+---------+---------+---------+---------+----------
v30   0         0         0         0         0
v50   1263      1262      1         1         1
v150  0         0         0         0         0
v200  0         0         0         0         0
```

This example shows PIM join and prune traffic statistics.

```
device(config)# show ip pim traffic join-prune rx
Port  Packet    Join      Prune     Avg Aggr  Last Aggr
-----+---------+---------+---------+---------+----------
      Rx        Rx        Rx        Rx        Rx
-----+---------+---------+---------+---------+----------
v30   0         0         0         0         0
v50   1260      1260      0         1         1
v150  0         0         0         0         0
v200  0         0         0         0         0
```

This example shows PIM join and prune traffic statistics.

```
device(config)# show ip pim traffic join-prune tx
Port  Packet    Join      Prune     Avg Aggr  Last Aggr
-----+---------+---------+---------+---------+----------
      Tx        Tx        Tx        Tx        Tx
-----+---------+---------+---------+---------+----------
v30   0         0         0         0         0
v50   1264      1263      1         1         1
v150  0         0         0         0         0
v200  0         0         0         0         0
```

# History

| Release version | Command history |
|---|---|
| 8.0.20a | This command was modified to display, in the Err column, received Hello packets dropped on an interface because of an ACL to control neighbor access. |

# show ip pimsm-snooping cache

Displays the downstream PIM join/prune information for both source-path tree (SPT) and rendezvous-point tree (RPT).

## Syntax

**show ip pimsm-snooping cache** [ **vlan** *vlan-id* ] *ip-address* [ **resources** ]

## Parameters

*ip-address*
Specifies the IP address.

**vlan** *vlan-id*
Specifies snooping for a VLAN.

**resources**
Specifies PIM SM snooping resources.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **show ip pimsm-snooping cache** command to check and verify the outgoing interfaces (OIF)s added by pimsm-snooping module.

## Command Output

The **show ip pimsm-snooping cache** command displays the following information:

| Output field | Description |
|---|---|
| SG | (s,g) downstream fsm state for SPT. |
| G | (*,g) downstream fsm state for RPT |

The **show ip pimsm-snooping cache** command displays the following information only when multi-chassis trunking (MCT) is enabled on the VLAN:

| Output field | Description |
|---|---|
| CCEP | Cluster client edge port |
| CEP | Cluster edge port |
| Remote/Local | Join/Prune received on MCT peer or local |

# Examples

```
Device1#show ip pimsm-snooping cache
 OIF Info:
 TR - OIF Belongs to Trunk/LAG, Primary port is displayed
 SG - (s,g) downstream fsm state:
 G - (*,g) downstream fsm state:
     NI : No Info, J : Join, PP : Prune Pending, CLEAN : cleanup in progress
 RPT - (s,g,rpt) downstream fsm state:
     NI : No Info, P : Pruned, PP : Prune Pending, Px : Temp step in (*,G)
     join processing, PPx : Temp State in (*,G) processing, CLEAN : cleanup
     in progress.

PIMSM Snoop cache for vlan 700, has 20 cache
1       (* 226.0.0.1) Up Time: 00:47:05
         OIF: 1
         e1/1/6 G : J(173) ET: 210, Up Time: 00:47:05 , CEP, Local

2       (80.1.1.9 226.0.0.1) Up Time: 00:47:04
         OIF: 1
         e1/1/6 SG : J(178) ET: 210, Up Time: 00:47:04 , CEP, Local
.....
.....
<output turncated>
...
...
9       (* 226.0.0.9) Up Time: 00:50:11
         OIF: 1
         e1/1/6 G : J(162) ET: 210, Up Time: 00:50:11 , CEP, Local

10      (* 226.0.0.10) Up Time: 00:50:11
         OIF: 1
         e1/1/6 G : J(167) ET: 210, Up Time: 00:50:11 , CEP, Local
```

The following example filters out sg-entries.

```
Device2#show ip pimsm-snooping cache sg-entries
 OIF Info:
 TR - OIF Belongs to Trunk/LAG, Primary port is displayed
 SG - (s,g) downstream fsm state:
 G - (*,g) downstream fsm state:
     NI : No Info, J : Join, PP : Prune Pending, CLEAN : cleanup in progress
 RPT - (s,g,rpt) downstream fsm state:
     NI : No Info, P : Pruned, PP : Prune Pending, Px : Temp step in (*,G)
     join processing, PPx : Temp State in (*,G) processing, CLEAN : cleanup
     in progress.

PIMSM Snoop cache for vlan 700, has 20 cache
1       (80.1.1.9 226.0.0.1) Up Time: 00:50:20
         OIF: 1
         e1/1/6 SG : J(162) ET: 210, Up Time: 00:50:20 , CEP, Local

2       (80.1.1.9 226.0.0.2) Up Time: 00:50:18
         OIF: 1
         e1/1/6 SG : J(161) ET: 210, Up Time: 00:50:18 , CEP, Local
....
....
<output truncated>
....
....
9       (80.1.1.9 226.0.0.9) Up Time: 00:50:19
         OIF: 1
         e1/1/6 SG : J(158) ET: 210, Up Time: 00:50:19 , CEP, Local

10      (80.1.1.9 226.0.0.10) Up Time: 00:50:19
         OIF: 1
         e1/1/6 SG : J(157) ET: 210, Up Time: 00:50:19 , CEP, Local
```

The following example filters out g-entries.

```
Device#show ipv6 pimsm-snooping cache g-entries
 OIF Info:
 TR - OIF Belongs to Trunk/LAG, Primary port is displayed
 SG - (s,g) downstream fsm state:
 G - (*,g) downstream fsm state:
     NI : No Info, J : Join, PP : Prune Pending, CLEAN : cleanup in progress
 RPT - (s,g,rpt) downstream fsm state:
     NI : No Info, P : Pruned, PP : Prune Pending, Px : Temp step in (*,G)
     join processing, PPx : Temp State in (*,G) processing, CLEAN : cleanup
     in progress.

PIMSM Snoop cache for vlan 700, has 20 cache
1      (* ff1e::6:1) Up Time: 00:57:33
        OIF: 1
        e1/1/6 G : J(175) ET: 210, Up Time: 00:57:33 , CEP, Local

2      (* ff1e::6:2) Up Time: 00:57:09
        OIF: 1
        e1/1/6 G : J(178) ET: 210, Up Time: 00:57:09 , CEP, Local
....
....
<output truncated>
....
....
9      (* ff1e::6:9) Up Time: 00:57:08
        OIF: 1
        e1/1/6 G : J(168) ET: 210, Up Time: 00:57:08 , CEP, Local

10     (* ff1e::6:a) Up Time: 00:57:35
        OIF: 1
        e1/1/6 G : J(169) ET: 210, Up Time: 00:57:35 , CEP, Local
```

# show ip reverse-path-check

Displays the global unicast Reverse Path Forwarding settings.

## Syntax

**show ip reverse-path-check**

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show ip reverse-path-check** command displays the following information.

| Output field | Description |
|---|---|
| CLI config | The command line configured on the device after device bootup. |
| Current state | The mode set during device bootup. This takes effect only after reload. |

## Examples

The following example shows the uRPF settings on ICX 7750 devices.

```
device# show ip reverse-path-check
Global uRPF Settings:
CLI config : Enabled
Current State : Enabled
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |
| 8.0.40 | Removed show output for the ICX 6610 device. |

# show ip reverse-path-check interface

Displays unicast Reverse Path Forwarding settings at the interface level on ICX devices.

## Syntax

**show ip reverse-path-check interface**

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to display the interface level unicast Reverse Path Forward settings such as the uRPF mode and whether uRPF excludes the default route for uRPF source IP lookup. Use the **show ip interface ethernet** command to view details about the interface level RPF mode configuration.

## Command Output

The **show ip reverse-path-check interface** command displays the following information.

| Output field | Description |
|---|---|
| Interface | The interface number. |
| uRPF mode | The uRPF mode enabled. |
| uRPF exclude default | Yes specifies that the exclude default option is enabled, while No specifies that the exclude default option is not enabled on the interface. |

## Examples

The following example shows the interface level uRPF settings on ICX devices.

```
device# show ip reverse-path-check interface
-----------------------------------------------------------------------
 Interface           uRPF mode               uRPF Exclude default
-----------------------------------------------------------------------
 Eth 1/1/11          Strict                     No
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show ip rip

Displays RIP filters.

## Syntax

**show ip rip**

## Modes

Privileged-EXEC mode

## Command Output

The **show ip rip** command displays the following information:

| Output field | Description |
|---|---|
| RIP Summary area | Shows the current configuration of RIP on the device. |
| Static metric | Shows the static metric configuration. "Not defined" means the route map has not been distributed. |
| OSPF metric | Shows what OSPF route map has been applied. |
| Neighbor Filter Table area | |
| Index | The filter number. You assign this number when you configure the filter. |
| Action | The action the device takes for RIP route packets to or from the specified neighbor: |
| | deny - If the filter is applied to an interface's outbound filter group, the filter prevents the device from advertising RIP routes to the specified neighbor on that interface. If the filter is applied to an interface's inbound filter group, the filter prevents the device from receiving RIP updates from the specified neighbor. |
| | permit - If the filter is applied to an interface's outbound filter group, the filter allows the device to advertise RIP routes to the specified neighbor on that interface. If the filter is applied to an interface's inbound filter group, the filter allows the device to receive RIP updates from the specified neighbor. |
| Neighbor IP Address | The IP address of the RIP neighbor. |

# Examples

The following example shows the current configuration of RIP on a device with a neighbor filter table configured to deny routes from source IP address 10.11.222.25.

```
device# show ip rip
RIP Summary
Default port 520
Administrative distance is 120
Updates every 30 seconds, expire after 180
Holddown lasts 180 seconds, garbage collect after 120
Last broadcast 29, Next Update 27
Need trigger update 0, Next trigger broadcast 1
Minimum update interval 25, Max update Offset 5
Split horizon is on; poison reverse is off
Import metric 1
Prefix List, Inbound : block_223
Prefix List, Outbound : block_223
Route-map, Inbound : Not set
Route-map, Outbound : Not set
Redistribute: CONNECTED Metric : 0 Routemap : Not Set

RIP Neighbor Filter Table
  Index   Action   Neighbor IP Address
    1     deny     10.11.222.55
    5     permit   any
```

# show ip rip interface

Displays RIP filters for a specific interface.

## Syntax

**show ip rip interface** [ **ethernet** *unit*/*slot*/*port* | **lag** *number* | **ve** *number* ]

## Parameters

**ethernet** *unit* **/** *slot* **/** *port*

Designates an Ethernet interface for which RIP filters are displayed.

**lag** *number*

Designates the LAG for which RIP filters are displayed.

**ve** *number*

Designates a virtual Ethernet interface for which RIP filters are displayed.

## Modes

Privileged EXEC mode

## Command Output

The **show ip rip interface** command displays the following information:

| Output field | Description |
|---|---|
| RIP mode: Version x | Specifies RIP version 1, version 2, or version 1-2 compatible. |
| Running: True/False | Indicates whether RIP protocol is active on the interface. |
| Route summarization | Indicates whether route summarization is enabled or disabled. |
| Split horizon is on/off; poison reverse is on/off | Indicates whether split horizon or poison reverse is enabled. |
| Default routes | Indicates whether default routes are accepted or not. |
| Metric-offset, Inbound | Indicates whether a value has been added to the metric for incoming (learned) routes. |
| Metric-offset, Outbound | Indicates whether a value has been added to the metric for outgoing (advertised) routes. |
| Prefix List, Inbound | Indicates whether a prefix list is applied to incoming routes. |
| Prefix List, Outbound | Indicates whether a prefix list is applied to outgoing routes. |
| Route-map, Inbound | Indicates whether a route-map is applied to incoming routes. |
| Route-map, Outbound | Indicates whether a route-map is applied to outgoing routes. |
| RIP Sent/Receive packet statistics | Provides number of requests and responses sent or received. |
| RIP Error packet statistics | Provides number of error packets by category: Rejected, Version, Response format, Address family, Metric, or Request format. |

# Examples

The following sample output shows that Ethernet interface 1/1/1 is running RIP Version 2 without prefix lists or route-maps and is adding 1 to the metric for learned RIP routes.

```
device# show ip rip interface ethernet 1/1/1
Interface e 1/1/1
RIP Mode : Version2 Running: TRUE
Route summarization disabled
Split horizon is on; poison reverse is off
Default routes not accepted
Metric-offset, Inbound 1
Metric-offset, Outbound 0
Prefix List, Inbound : Not set
Prefix List, Outbound : Not set
Route-map, Inbound : Not set
Route-map, Outbound : Not set
RIP Sent/Receive packet statistics:
Sent : Request 2 Response 34047
Received : Total 123473 Request 1 Response 123472 UnRecognised 0
RIP Error packet statistics:
Rejected 0 Version 0 RespFormat 0 AddrFamily 0
Metric 0 ReqFormat 0
```

# show ip rip route

Displays RIP route information for a device or a specific interface.

## Syntax

**show ip rip route** [ *ip-address* | *ip-address / L* ]

## Parameters

*ip-address*

Specifies the IP address, in the format A.B.C.D, for which RIP routes are displayed.

ip-address / L

Specifies the IP address prefix and mask, in the format A.B.C.D/L, where "L" is the mask length. Information is displayed for IP addresses matching the mask.

## Modes

Privileged EXEC mode

## Command Output

The **show ip rip route** command displays the following information:

| Output field | Description |
|---|---|
| RIP Routing Table - nn entries | Indicates the number of routes in the device's routing table. |
| RIP route designation | Designates each route by CIDR designation, originating IP address, and interface. |
| RIP route settings | For each designated route, indicates protocol, metric setting, tag, and non-default timer settings. |

## Examples

The following example shows RIP route information for the device.

```
device# show ip rip route
RIP Routing Table - 474 entries:
1.1.1.1/32, from 169.254.30.1, e 1/1/23 (820)
RIP, metric 4, tag 0, timers: aging 13
1.1.2.1/32, from 169.254.50.1, e 1/3/1 (482)
RIP, metric 3, tag 0, timers: aging 42
1.1.6.1/32, from 169.254.100.1, ve 101 (413)
RIP, metric 2, tag 0, timers: aging 42
169.254.40.0/24, from 192.168.1.2, e 1/1/1 (1894)
RIP, metric 3, tag 0, timers: aging 14
169.254.50.0/24, from 192.168.1.2, e 1/1/1 (1895)
RIP, metric 4, tag 0, timers: aging 14
169.254.100.0/24, from 192.168.1.2, e 1/1/1 (2040)
RIP, metric 2, tag 0, timers: aging 14
169.254.101.0/30, from 192.168.1.2, e 1/1/1 (2105)
223.229.32.0/31, from 169.254.50.1, e 1/3/1 (818)
RIP, metric 2, tag 0, timers: aging 21
```

# show ip route

Displays the IP route table information.

## Syntax

**show ip route** [ **vrf** *vrf-name* ] [ *ip-addr* | *num* | **bgp** | **direct** | **ospf** | **rip** | **static** | **summary** ]

## Parameters

**vrf** *vrf-name*
Displays VRF routes.

*ip-addr*
Displays information for the subnet mask.

*num*
Displays route starting from index.

**bgp**
Displays BGP routes.

**direct**
Displays directly attached routes.

**ospf**
Displays OSPF routes.

**rip**
Displays RIP routes.

**static**
Displays static IP routes.

**summary**
Displays route summary.

## Modes

User EXEC mode

## Command Output

The **show ip route** command displays the following information:

| Output field | Description |
|---|---|
| Destination | The destination network of the route. |
| Cost | The route's cost. |
| Type | The route type, which can be one of the following:<br>• **B** - The route was learned from BGP.<br>• **D** - the destination is directly connected to this Ruckus device. |

| Output field | Description |
|---|---|
| | • **R**- The route was learned from RIP.<br>• **S** - The route is a static route.<br>• ***** - The route is a candidate default route.<br>• **O** - The route is an OSPF route. Unless you use the OSPF option to display the route table, 'O' is used for all OSPF routes. If you do not use the OSPF option, the following type codes are used:<br>– **O** - OSPF intra area route (within the same area.)<br>– **IA** - The route is an OSPF inter area route (a route that passes from one area in another area.)<br>– **E1** - The route is an OSPF external type 1 route.<br>– **E2** - The route is an external type 2 route. |

# Examples

The following example shows the **show ip route** command:

```
device# show ip route
Total number of IP routes: 2
Type Codes - B:BGP D:Connected O:OSPF R:RIP S:Static; Cost - Dist/Metric
BGP  Codes - i:iBGP e:eBGP
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2
        Destination        Gateway         Port          Cost          Type    Uptime
1       0.0.0.0/0          10.25.224.1     e mgmt1       254/1         S       8d6h

2       10.25.224.0/24     DIRECT          e mgmt1       0/0           D       6h39
```

# show ip source-guard

Displays the learned IP addresses for IP Source Guard ports.

## Syntax

**show ip source-guard ethernet** *stack-id/slot/port*

## Parameters

**ethernet***stack-id/slot/port*
Specifies the Ethernet interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show ip source-guard** command displays the following information:

| Output field | Description |
|---|---|
| Interface | Displays the interface number for source guard entries learnt or configured statically. |
| Type | Displays the interface type - IP. |
| Filter mode | Displays the filter mode - active or inactive. |
| IP-address | The dynamically learned or statically configured address. |
| VLAN | Specifies the VLAN number. |
| Static | All the static source guard entries configured are populated as "Yes". |

## Examples

The following output displays the learned IP addresses for IP Source Guard ports.

```
device# show ip source-guard e 1/1/48
Total IP Source Guard entries on port 1/1/48: 33
No     Interface           Type   Flter-mode      IP-address      Vlan   Static
--     ---------           ----   ----------      ----------      ----   -------
1      1/1/9*4/1/39         ip     active         15.15.15.127     1       Yes
2      1/1/9*4/1/39         ip     active         15.15.15.9       1       No
3      1/1/9*4/1/39         ip     active         15.15.15.10      1       No
4      1/1/9*4/1/39         ip     active         15.15.15.11      1       No
5      1/1/9*4/1/39         ip     active         15.15.15.12      1       No
6      1/1/9*4/1/39         ip     active         15.15.15.13      1       No
7      1/1/9*4/1/39         ip     active         15.15.15.14      1       No
8      1/1/9*4/1/39         ip     active         15.15.15.15      1       No
9      1/1/9*4/1/39         ip     active         15.15.15.16      1       No
10     1/1/9*4/1/39         ip     active         15.15.15.17      1       No
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | The output of this command was modified for static source guard entries. |

# show ip ssh

Displays Secure Shell (SSH) connection session details.

## Syntax

**show ip ssh** [ **config** | **rekey statistics** ]

\

## Parameters

**config**

Displays the SSH configuration details.

**rekey statistics**

Displays the SSH rekey statistics information.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show ip ssh** command displays the following information:

| Output field | Description |
|---|---|
| Inbound | Connections listed under this heading are inbound. |
| Outbound | Connections listed under this heading are outbound. |
| Connection | The SSH connection ID. |
| Version | The SSH version number. |
| Encryption | The encryption method used for the connection. |
| Username | The username for the connection. |
| HMAC | The HMAC version. |
| Server Hostkey | The type of server host key. This can be DSA or RSA. |
| IP Address | The IP address of the SSH client. |
| SSH-v2.0 enabled | Indicates that SSHv2 is enabled. |
| hostkey | Indicates that at least one host key is on the device. It is followed by a list of the host key types and module sizes. |

The **show ip ssh config** command displays the following information:

| Output field | Description |
|---|---|
| SSH server | SSH server is enabled or disabled. |

| Output field | Description |
|---|---|
| SSH port | SSH port number. |
| Host Key | Host key. |
| Encryption | The encryption used for the SSH connection. The following values are displayed when AES only is enabled:<br>• AES-256, AES-192, and AES-128 indicate the different AES methods used for encryption.<br>• 3-DES indicates 3-DES algorithm is used for encryption |
| Permit empty password | Empty password login is allowed or not allowed. |
| Authentication methods | The authentication methods used for SSH. The authentication can have one or more of the following values:<br>• Password: Indicates that you are prompted for a password when attempting to log in to the device.<br>• Public-key: Indicates that DSA or RSA challenge-response authentication is enabled.<br>• Interactive: Indicates the interactive authentication is enabled. |
| Authentication retries | The number of authentication retries. This number can be from 1 through 5. |
| Login timeout (seconds) | SSH login timeout value in seconds. This can be from 0 through 120. |
| Idle timeout (minutes) | SSH idle timeout value in minutes. This can be from 0 through 240. |
| Strict management VRF | Strict management VRF is enabled or disabled. |
| SCP | SCP is enabled or disabled. |
| SSH IPv4 clients | The list of IPv4 addresses to which SSH access is allowed. The default is "All". |
| SSH IPv6 clients | The list of IPv6 addresses to which SSH access is allowed. The default is "All". |
| SSH IPv4 access-group | The IPv4 ACL used to permit or deny access using SSH. |
| SSH IPv6 access-group | The IPv6 ACL used to permit or deny access using SSH. |
| Client Rekey | The SSH rekey interval configured for the client, in minutes and maximum data. |
| Server Rekey | The SSH rekey interval configured for the server, in minutes and maximum data. |

# Examples

The following example displays sample output of the **show ip ssh** command.

```
device# show ip ssh
Connection  Version  Encryption  Username  HMAC       Server Hostkey  IP Address
Inbound:
1           SSH-2    3des-cbc    Raymond   hmac-sha1  ssh-dss         10.120.54.2
Outbound:
6           SSH-2    aes256-cbc  Steve     hmac-sha1  ssh-dss         10.37.77.15
SSH-v2.0 enabled; hostkey: DSA(1024), RSA(2048)
```

The following example displays sample output of the **show ip ssh config** command.

```
device# show ip ssh config
SSH server                  : Disabled
SSH port                    : tcp\22
Host Key                    :
Encryption                  : aes256-cbc, aes192-cbc, aes128-cbc, aes256-ctr, aes
192-ctr, aes128-ctr, 3des-cbc
Permit empty password       : No
Authentication methods      : Password, Public-key, Interactive
Authentication retries      : 3
Login timeout (seconds)     : 120
Idle timeout (minutes)      : 0
Strict management VRF       : Disabled
SCP                         : Enabled
SSH IPv4 clients            : All
SSH IPv6 clients            : All
SSH IPv4 access-group       :
SSH IPv6 access-group       :
SSH Client Keys             : RSA(0)
Client Rekey                : 200 Minute, 0 KB
Server Rekey                : 250 Minute, 0 KB
```

The following example displays sample output of the **show ip ssh rekey statistics** command.

```
device# show ip ssh rekey statistics
SSH Server Rekey Statistics:
1     Time :   24 Sec,   Data :     996632 Bytes
2     closed
3     closed
4     closed
5     closed
SSH Client Rekey Statistics:
6     Time :  596 Sec, Data :    2999556 Bytes
7     closed
8     closed
9     closed
10    closed
11    closed
12    closed
13    closed
14    closed
15    closed
16    closed
17    closed
18    closed
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was modified to add **rekey statistics** option. |

# show ip ssl (FIPS)

Displays SSL connection details.

## Syntax

**show ip ssl** [ **certificate** | **client-certificate** | **profile** ]

## Parameters

**certificate**
Displays the SSL certificate details.

**client-certificate**
Displays SSL client certificate details.

**profile**
Displays SSL profile details.

## Modes

Privileged EXEC mode (with FIPS enabled)

Global configuration mode (with FIPS enabled)

## Examples

The following example displays the output of the **show ip ssl** command.

```
device(config)# show ip ssl
Session Protocol Source IP      Source Port  Remote IP      Remote Port
1       TLS_1_2  10.20.157.102  634          10.25.105.201  60892
```

The following example displays output for the **show ip ssl profile** command.

```
device# show ip ssl profile
SSL Profile Information:
****************************************

Profile Name   : LINUX-END
Trustpoint Name: TLS-LINUX
Remote Domain  : LINUX-END
****************************************

Profile Name   : LINUX-END-SAN
Trustpoint Name: TLS-LINUX
Remote Domain  : LINUX-END-SAN
****************************************

Profile Name   : p1
Trustpoint Name: TLS-LINUX
Remote Domain  : example.com
****************************************

Profile Name   : p2
Trustpoint Name: TLS-LINUX
Remote Domain  : ruckus.com
****************************************

Profile Name   : p3
Trustpoint Name: auto
Remote Domain  : gss.example.com
****************************************

Profile Name   : p4
Trustpoint Name: TLS-LINUX
Remote Domain  : *.example.com
****************************************

Profile Name   : p5
Trustpoint Name: TLS-LINUX
Remote Domain  : LINUX-SERVER-SAN
****************************************

Profile Name   : p6
Trustpoint Name: TLS-LINUX
Remote Domain  : LINUX-SERVER-CC
****************************************
```

The following example displays SSL certificate details.

```
device(config)# show ip ssl certificate
Trusted Certificates:
 Dynamic:
 Index 0:
  Signature Algorithm: sha256WithRSAEncryption
  Issuer:
   CN: 10.25.105.201
  Validity:
   Not Before: 2014 Aug 22 05:12:45
   Not After : 2017 Aug 21 05:12:45
  Subject:
   CN: 10.25.105.201
  X509v3 extensions:
   X509v3 Subject Alternative Name:
    IP Address: 10.25.105.201
  Signature:
   12:ec:41:d8:01:45:61:ce:cf:7e:80:de:a6:7c:a7:2e:01:7f:
   42:27:22:1d:ac:a2:47:c5:0d:4f:e3:68:24:de:bf:50:40:65:
   25:8c:30:bd:ff:a7:d0:21:73:d2:ba:5e:67:42:1f:bb:97:4a:
   d9:1d:c3:ca:31:c4:59:10:79:d1:42:f4:b6:1a:b0:98:4e:a8:
   ef:e2:a2:98:c3:14:16:63:50:02:a0:18:9c:7a:e3:17:39:0d:
   b7:30:ab:23:9f:63:bd:0f:9e:d8:67:b0:fe:ec:3b:fa:4c:f4:
   3d:34:e2:99:0e:99:24:ec:93:fb:8a:e5:4a:bf:74:d6:ff:91:
   0a:dc:fb:b9:4f:91:5d:d4:f6:77:23:eb:ec:eb:3a:62:08:e1:
   a6:ea:a8:52:b6:39:62:db:29:fa:61:1d:fd:d5:02:31:04:73:
   50:ad:de:41:54:a5:e2:96:2d:9c:f4:68:b2:68:05:bb:39:47:
   ee:74:89:a2:8c:30:f0:f9:d7:d5:4b:3b:e2:95:6f:82:61:a3:
   c2:79:4c:f2:11:56:f8:2f:cc:fc:2b:4b:cb:3b:54:59:f0:8b:
   5b:70:e1:27:c3:57:25:eb:35:c6:07:ea:6d:0b:34:04:95:81:
   35:e6:64:c6:b8:72:e8:24:18:bd:ca:90:99:74:45:44:85:71:
   9e:7f:13:96:
```

The following example displays SSL client certificate information.

```
device(config)# show ip ssl client-certificate
SSL Client Certificates:
**********************************************************
Trustpoint Name: TLS-LINUX
  Signature Algorithm: sha256WithRSAEncryption
  Issuer:
   CN: ROOTCA-CC
  Validity:
   Not Before: 2017 Nov  6 18:24:18
   Not After : 2018 Nov 16 18:24:18
  Subject:
   CN: DUTFIPSCC
  Signature:
   76:24:88:c3:07:f3:37:e0:c7:06:17:a8:39:03:ad:ad:d8:ee:
   f0:76:ac:4f:5b:08:d6:3b:0c:3d:36:b4:1e:ac:cd:b7:76:2b:
   a7:7e:22:94:63:56:5b:88:64:3a:62:a8:80:c7:b4:57:8d:a8:
   51:1c:34:7c:b4:27:d2:92:9f:f2:f8:26:24:de:a6:b9:e5:93:
   ee:08:47:cc:6a:09:03:62:bf:06:2e:14:c0:51:d8:0d:aa:a5:
   4e:b4:1e:91:2c:05:f8:87:a1:48:6c:4c:0b:4e:02:7f:b7:8f:
   6e:1e:a8:9b:00:e0:a8:62:56:5f:25:dd:49:e1:76:42:0f:ea:
   3f:79:43:06:eb:76:53:48:1c:4c:2d:ef:04:f7:1b:96:8d:31:
   3b:ce:d5:33:8f:7c:2e:88:a5:1c:87:ed:c1:99:71:42:c5:62:
   08:46:a4:d7:a3:54:0d:b1:0f:29:5d:1a:fa:9e:02:f1:de:d9:
   89:3a:44:a8:31:0c:85:76:7e:ad:fb:09:6e:af:9c:7f:2e:57:
   27:b1:8a:9c:d3:a6:b1:67:ca:7f:70:26:0b:e2:87:3d:23:ac:
   1c:e8:8f:02:eb:1d:b3:af:0a:ac:81:3b:73:58:8a:79:1f:7e:
   c2:9f:8b:e1:73:dd:fb:76:33:20:84:69:cb:5c:82:cd:4c:8f:
   c1:98:9f:ac:
**********************************************************
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command changes the **certificate-client** parameter to **client-certificate**. |

# show ip static mroute

Displays information for configured multicast routes.

## Syntax

**show ip static mroute** [ **vrf** *vrf-name* ] *ip-subnet mask*]

## Parameters

**vrf** *vrf-name*

Specifies an optional VRF route.

*ip-subnet mask*

Specifies an IP address and an optional address mask.

## Modes

Privileged EXEC mode

## Usage Guidelines

Only resolved and best static mroutes are added to the mRTM table. These routes are prefixed with an asterisk in the output from the **show ip static mroute** command.

## Examples

The following example displays information for configured multicast routes:

```
Device(config)# show ip static mroute
IP Static Routing Table - 2 entries:
 IP Prefix          Next Hop        Interface  Dis/Metric/Tag  Name
*20.20.20.0/24      220.220.220.1   -          1/1/0
20.20.20.0/24       50.50.50.2      -          1/2/0
21.21.21.0/24       1.2.3.4         -          1/1/0
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# show ip static-arp

Displays the static ARP entries along with static inspect ARP entries.

## Syntax

**show ip static-arp**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

The display index for static inspect ARP entries is not be displayed in the command output.

## Command Output

The **show ip static-arp** command displays the following information:

| Output field | Description |
|---|---|
| Static ARP table size | The maximum number of static ARP entries that can be configured. The default value is 512, and can be changed to 1024 using the **max-static-inspect-arp-entries** command. |

## Examples

The following example displays the static ARP.

```
device# show ip static-arp
Static ARP table size: 512, configurable from 512 to 1024
Index   IP Address          MAC Address       Port
1       207.95.6.111       0800.093b.d210     1/1/1
3       207.95.6.123       0800.093b.d211     1/1/1
-       1.1.1.1           0800.0000.0001     Invalid
```

## History

| Release version | Command history |
|---|---|
| 08.0.30b | This command was modified. The output does not display the index for static inspect ARP entries. |

# show ip traffic

Displays IP traffic statistics.

## Syntax

**show ip traffic**

## Modes

User EXEC mode

## Command Output

The **show ip traffic** command displays the following information:

| Output field | Description |
|---|---|
| IP statistics | |
| received | The total number of IP packets received by the device. |
| sent | The total number of IP packets originated and sent by the device. |
| fragmented | The total number of IP packets fragmented by this device to accommodate the MTU of this device or of another device. |
| reassembled | The total number of fragmented IP packets that this device re-assembled. |
| bad header | The number of IP packets dropped by the device due to a bad packet header. |
| no route | The number of packets dropped by the device because there was no route. |
| unknown proto | The number of packets dropped by the device because the value in the Protocol field of the packet header is unrecognized by this device. |
| no buffer | This information is used by Ruckus customer support. |
| other errors | The number of packets that this device dropped due to error types other than the types listed above. |
| ICMP statistics<br>The ICMP statistics are derived from RFC 792, "Internet Control Message Protocol", RFC 950, "Internet Standard Subnetting Procedure", and RFC 1256, "ICMP Router Discovery Messages". Statistics are organized into Sent and Received. The field descriptions below apply to each. | |
| total | The total number of ICMP messages sent or received by the device. |
| errors | This information is used by Ruckus customer support. |
| unreachable | The number of Destination Unreachable messages sent or received by the device. |
| time exceed | The number of Time Exceeded messages sent or received by the device. |
| parameter | The number of Parameter Problem messages sent or received by the device. |
| source quench | The number of Source Quench messages sent or received by the device. |
| redirect | The number of Redirect messages sent or received by the device. |
| echo | The number of Echo messages sent or received by the device. |
| echo reply | The number of Echo Reply messages sent or received by the device. |
| timestamp | The number of Timestamp messages sent or received by the device. |
| timestamp reply | The number of Timestamp Reply messages sent or received by the device. |

| Output field | Description |
|---|---|
| addr mask | The number of Address Mask Request messages sent or received by the device. |
| addr mask reply | The number of Address Mask Replies messages sent or received by the device. |
| irdp advertisement | The number of ICMP Router Discovery Protocol (IRDP) Advertisement messages sent or received by the device. |
| irdp solicitation | The number of IRDP Solicitation messages sent or received by the device. |
| UDP statistics | |
| received | The number of UDP packets received by the device. |
| sent | The number of UDP packets sent by the device. |
| no port | The number of UDP packets dropped because the packet did not contain a valid UDP port number. |
| TCP statistics<br>The TCP statistics are derived from RFC 793, "Transmission Control Protocol".<br>input errors<br>This information is used by Ruckus customer support. | |
| current active tcbs | The number of TCP Control Blocks (TCBs) that are currently active. |
| tcbs allocated | The number of TCBs that have been allocated. |
| tcbs freed | The number of TCBs that have been freed. |
| tcbs protected | This information is used by Ruckus customer support. |
| active opens | The number of TCP connections opened by this device by sending a TCP SYN to another device. |
| passive opens | The number of TCP connections opened by this device in response to connection requests (TCP SYNs) received from other devices. |
| failed attempts | This information is used by Ruckus customer support. |
| active resets | The number of TCP connections this device reset by sending a TCP RESET message to the device at the other end of the connection. |
| passive resets | The number of TCP connections this device reset because the device at the other end of the connection sent a TCP RESET message. |
| input errors | This information is used by Ruckus customer support. |
| in segments | The number of TCP segments received by the device. |
| out segments | The number of TCP segments sent by the device. |
| retransmission | The number of segments that this device retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment. |

# Examples

The following is sample output from the **show ip traffic** command.

```
device# show ip traffic
IP Statistics
  875327 received, 120 sent, 0 forwarded
  0 filtered, 0 fragmented, 0 reassembled, 0 bad header
  0 no route, 0 unknown proto, 0 no buffer, 32124 other errors

ARP Statistics
  553661 total recv, 538907 req recv, 78 req sent, 137 rep sent
  0 pending drop, 0 invalid source, 0 invalid dest
  0 mis-match dst-mac, 0 mis-match ip addr, 0 mis-match src-mac

ICMP Statistics
Received:
  1 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 1 echo,
  0 echo reply, 0 timestamp, 0 timestamp reply, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation
Sent:
  1 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  1 echo reply, 0 timestamp, 0 timestamp reply, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation

UDP Statistics
  42046 received, 119 sent, 41930 no port, 0 input errors

TCP Statistics
  0 active opens, 0 passive opens, 0 failed attempts
  0 active resets, 0 passive resets, 0 input errors
  0 in segments, 0 out segments, 0 retransmission
```

# show ip tunnel traffic

Displays the link status of the tunnel and the number of keepalive packets received and sent on the tunnel.

## Syntax

**show ip tunnel traffic**

## Modes

User EXEC mode

## Command Output

The **show ip tunnel traffic** command displays the following information:

| Output field | Description |
|---|---|
| Tunnel Status | Indicates whether the tunnel is up or down. Possible values are:<br>• Up/Up - The tunnel and line protocol are up.<br>• Up/Down - The tunnel is up and the line protocol is down.<br>• Down/Up - The tunnel is down and the line protocol is up.<br>• Down/Down - The tunnel and line protocol are down. |
| Packet Received | The number of packets received on the tunnel since it was last cleared by the administrator. |
| Packet Sent | The number of packets sent on the tunnel since it was last cleared by the administrator. |
| KA recv | The number of keepalive packets received on the tunnel since it was last cleared by the administrator. |
| KA sent | The number of keepalive packets sent on the tunnel since it was last cleared by the administrator. |

## Examples

The following output from the **show ip tunnel traffic** command displays the link status of the tunnel and the number of keepalive packets received and sent on the tunnel.

```
device# show ip tunnel traffic
IP GRE Tunnels
Tunnel    Status         Packet Received    Packet Sent    KA recv      KA sent
1         up/up             362                  0           362          362
3         up/up             0                    0           0            0
10        down/down         0                    0           0            0
```

# show ip vrrp

Displays information about IPv4 Virtual Router Redundancy Protocol (VRRP) sessions.

## Syntax

**show ip vrrp** [ **brief** ]

**show ip vrrp** [ **ethernet** *unit**/**slot**/**port* | **ve** *num* ]

**show ip vrrp** [ **statistics** [ **ethernet** *unit**/**slot**/**port* | **ve** *num* ] ]

**show ip vrrp** [ **ve** *num* [ **vrid** *VRID* ] ]

**show ip vrrp** [ **vrid** *VRID* [ **ethernet** *unit**/**slot**/**port* | **ve** *num* ] ]

## Parameters

**brief**
Displays summary information about the VRRP session.

**ethernet** *unit**/**slot**/ port*
Displays IPv4 VRRP information only for the specified port. A forward slash "/" must be entered between the unit, slot, and port numbers.

**statistics**
Displays statistical information about the VRRP session.

**ve** *num*
Displays IPv4 VRRP information only for the specified virtual Ethernet port.

**vrid** *VRID*
Displays IPv4 VRRP information only for the specified virtual-group ID.

## Modes

User EXEC mode

## Usage Guidelines

Use this command to display information about IPv4 VRRP sessions, either in summary or full-detail format. You can also specify a virtual group or interface for which to display output.

This command supports IPv4 VRRP. You can modify or redirect the displayed information by using the default Linux tokens (|, >).

## Command Output

The **show ip vrrp** command displays the following information.

| Output field | Description |
|---|---|
| Total number of VRRP routers defined | The total number of virtual routers configured and currently running on this Ruckus ICX device. For example, if the device is running VRRP-E, the total applies only to VRRP-E routers. |
| Interface | The interface on which VRRP or VRRP-E is configured. If VRRP or VRRP-E is configured on multiple interfaces, information for each interface is listed separately. |
| VRID | The ID of the virtual router configured on this interface. If multiple virtual routers are configured on the interface, information for each virtual router is listed in a separate row. |
| Current Priority | The current VRRP or VRRP-E priority of this Ruckus device for the virtual router. |
| Flags Codes | Whether the backup preempt mode is enabled and which version of VRRP is enabled. If the backup preempt mode is enabled, this field contains a "P". If the mode is disabled, this field is blank.<br>• P:Preempt<br>• 2:V2—VRRP Version 2<br>• 3:V3—VRRP Version 3<br>• S:Short-Path-Fwd—Short-path forwarding is enabled |
| State | This device's VRRP state for the virtual router. The state can be one of the following:<br>• Init—The virtual router is not enabled (activated). If the state remains Init after you activate the virtual router, make sure that the virtual router is also configured on the other routers and that the routers can communicate with each other.<br>If the state is Init and the mode is incomplete, make sure that you have specified the IP address for the virtual router.<br>• Backup—This device is a backup for the virtual router.<br>• Master—This device is the master for the virtual router. |
| Master IP Address | The IP address of the router interface that is currently the Master for the virtual router. If the IP address is assigned on this device, "Local" is displayed here. |
| Backup IP Address | The IP addresses of the router interfaces that are currently backups for the virtual router. If the IP address is not known in th routing table, "Unknown" is displayed here. |
| Virtual IP Address | The virtual IP address that is being backed up by the virtual router. |

# Examples

The following example displays VRRP session information in summary format.

```
device(config)# show ip vrrp brief

Total number of VRRP routers defined: 2
Flags Codes - P:Preempt 2:V2 3:V3 S:Short-Path-Fwd
Inte- VRID  Current  Flags    State   Master IP Backup IP  Virtual IP
rface       Priority                 Address   Address    Address
-------------------------------------------------------------------------
1/1/1  10    255      P2-     Master  Local     Unknown    10.30.30.2
1/1/3  13    100      P2-     Master  Local     Unknown    10.13.13.3
```

The following example displays IPv4 VRRP configuration information about VRID 1.

```
device# show ip vrrp vrid 1

Interface 1/1/1
----------------
auth-type no authentication
VRID 1 (index 1)
interface 1/1/1
state master
administrative-status enabled
version v2
mode owner
virtual mac aaaa.bbbb.cccc (configured)
priority 255
current priority 255
track-priority 2
hello-interval 1 sec
backup hello-interval 6
```

# show ip vrrp-extended

Displays information about IPv4 Virtual Router Redundancy Protocol Extended (VRRP-E) sessions.

## Syntax

**show ip vrrp-extended** [ **brief** ]

**show ip vrrp-extended** [ **ethernet** *unit*/*slot*/*port* | **ve** *num* ]

**show ip vrrp-extended** [ **statistics** [ **ethernet** *unit*/*slot*/*port* | **ve** *num* ] ]

**show ip vrrp-extended** [ **ve** *num* [ **vrid** *VRID* ] ]

**show ip vrrp-extended** [ **vrid** *VRID* [ **ethernet** *unit*/*slot*/*port* | **ve** *num* ] ]

## Parameters

**brief**
Displays summary information about the VRRP-E session.

**ethernet** *unit*/*slot*/ *port*
Displays IPv4 VRRP-E information only for the specified port. A forward slash "/" must be entered between the unit, slot, and port numbers.

**ve** *num*
Displays IPv4 VRRP-E information only for the specified virtual Ethernet port.

**statistics**
Displays statistical information about the VRRP-E session.

**vrid** *VRID*
Displays IPv4 VRRP-E information only for the specified virtual-group ID.

## Modes

User EXEC mode

## Usage Guidelines

Use this command to display information about IPv4 VRRP-E sessions, either in summary or full-detail format. You can also specify a virtual group or interface for which to display output.

This command supports IPv4 VRRP-E. You can modify or redirect the displayed information by using the default Linux tokens (|, >).

This command can be entered in any mode on the device.

## Command Output

The **show ip vrrp-extended** command displays the following information.

| Output field | Description |
|---|---|
| Total number of VRRP-E routers defined | The total number of virtual routers configured and currently running on this Ruckus ICX device. For example, if the device is running VRRP-E, the total applies only to VRRP-E routers. |
| Interface | The interface on which VRRP or VRRP-E is configured. If VRRP or VRRP-E is configured on multiple interfaces, information for each interface is listed separately. |
| VRID | The ID of the virtual router configured on this interface. If multiple virtual routers are configured on the interface, information for each virtual router is listed in a separate row. |
| Current Priority | The current VRRP or VRRP-E priority of this device for the virtual router. |
| Flags | Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains a "P". If the mode is disabled, this field is blank.<br>• P:Preempt 2:V2 3:V3<br>• 2: implies VRRP Version2<br>• 3: implies VRRP Version3 |
| State | This device's VRRP state for the virtual router. The state can be one of the following:<br>• Init—The virtual router is not enabled (activated). If the state remains Init after you activate the virtual router, make sure that the virtual router is also configured on the other routers and that the routers can communicate with each other.<br>If the state is Init and the mode is incomplete, make sure that you have specified the IP address for the virtual router.<br>• Backup—This device is a backup for the virtual router.<br>• Master—This device is the master for the virtual router. |
| Master IP Address | The IP address of the router interface that is currently the Master for the virtual router. If the IP address is assigned on this device, "Local" is displayed here. |
| Backup IP Address | The IP addresses of the router interfaces that are currently backups for the virtual router. If the IP address is not known in th routing table, "Unknown" is displayed here. |
| Virtual IP Address | The virtual IP address that is being backed up by the virtual router. |

# Examples

The following example displays summary information for a VRRP-E session.

```
device# show ip vrrp-extended brief

Total number of VRRP-E routers defined: 2
Flags Codes - P:Preempt 2:V2 3:V3 S:Short-Path-Fwd
Inte- VRID  Current  Flags    State   Master IP Backup IP  Virtual IP
rface       Priority                  Address   Address    Address
-------------------------------------------------------------------------
Ve 1  2     255      P2-      Master  Local     10.30.20.2 10.30.30.2
Ve 3  4     100      P2-      Backup  Local     10.30.20.2 10.30.30.2
```

The following example displays detailed information for a VRRP-E backup device.

```
device(config)# show ip vrrp-extended

Total number of vrrp-extended routers defined: 1
Interface v10
----------------
auth-type no authentication
VRID 10 (index 1)
interface v10
state backup
administrative-status enabled
mode non-owner(backup)
virtual mac 02e0.52a0.c00a
priority 50
current priority 50
track-priority 5
hello-interval 1 sec
backup hello-interval 60 sec
slow-start timer (configured) 30 sec
advertise backup disabled
dead-interval 3600 ms
preempt-mode true
virtual ip address 10.10.10.254
next hello sent in 1000ms
track-port 1/1/1 (up)
master router 10.10.10.4 expires in 3.1 sec
short-path-forwarding enabled
```

The following example displays IPv4 VRRP-E statistics. The “received vrrp-extended packets with unknown or inactive vrid” shows the number of packets that contain virtual router IDs that are not configured on the device or its interface.

```
device> show ip vrrp-extended statistics

Global VRRP-Extended statistics
-------------------------------
- received vrrp-extended packets with checksum errors = 0
- received vrrp-extended packets with invalid version number = 0
- received vrrp-extended packets with unknown or inactive vrid = 1480
Interface v10
----------------
VRID 1
- number of transitions to backup state = 1
- number of transitions to master state = 1
- total number of vrrp-extended packets received = 0
. received backup advertisements = 0
. received packets with zero priority = 0
. received packets with invalid type = 0
. received packets with invalid authentication type = 0
. received packets with authentication type mismatch = 0
. received packets with authentication failures = 0
. received packets dropped by owner = 0
. received packets with ip ttl errors = 0
. received packets with ip address mismatch = 0
. received packets with advertisement interval mismatch = 0
. received packets with invalid length = 0
- total number of vrrp-extended packets sent = 2004
. sent backup advertisements = 0
. sent packets with zero priority = 0
- received arp packets dropped = 0
- received proxy arp packets dropped = 0
- received ip packets dropped = 0
```

The following example displays IPv4 VRRP-E configuration information about VRID 1.

```
device# show ip vrrp-extended vrid 1

Interface 1/1/1
---------------
auth-type md5-authentication
VRID 1 (index 1)
interface 1/1/1
state master
administrative-status disabled
mode non-owner(backup)
virtual mac aaaa.bbbb.cccc (configured)
priority 100
current priority 100
track-priority 5
hello-interval 1 sec
backup hello-interval 60 sec
slow-start timer (configured) 30 sec
advertise backup disabled
dead-interval 0 ms
preempt-mode true
virtual ip address 10.20.1.100
short-path-forwarding disabled
```

The following example displays whether the VRRP-E hitless upgrade feature is enabled. This feature is used in conjunction with the short-path forwarding feature. In this example, the **activate backup** and the **short-path-forwarding** commands are enabled. Only partial output is displayed.

```
device# show ip vrrp-extended

Total number of VRRP-Extended routers defined: 1
Interface v10
 auth-type no authentication
 VRID 5
  state backup
  administrative-status enabled
.
.
.
  short-path-forwarding enabled
  activate-backup: enabled
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to add new output for the VRRP-E hitless upgrade feature. |

# show ipc_stats

Displays reliable Inter-process Communications (IPC) and dynamic queue statistics.

## Syntax

**show ipc_stats**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

NTP configuration mode

## Examples

The following is sample output from the **show ipc_stats** command.

```
device# show ipc_stats

Total available Hsync channel space = 1048580
Total available Appl channel space = 524292
Total number of application msgs in dyn queue = 0
Total number of hsync msgs in dyn queue = 0
Total number of rel sync msgs in dyn queue = 0
Total number of rx pkt msgs in standby dynamic queue = 0
Total number of rx pkt msgs in active dyn queue = 0
Total number of rx pkts relayed = 0
Total number of rx pkts received = 5686578
Total number of dyn-sync messages received so far = 3
Total number of rel-sync pending complete = 0
Total number of L3 baseline-sync packets = 655
Total number of packet drops in sync = 0
Is image_sync_in_progress? = 0
Total num of rx dyn queue drops = 0
Total num of jumbo corrupts = 0
Total number of messages in IP send queue = 0
```

# show ipsec card-utilization

Displays information about the utilization of the IPsec interface module that includes the administration status of the module and traffic statistics.

## Syntax

**show ipsec card-utilization**

## Modes

Privileged EXEC mode

## Usage Guidelines

Traffic utilization percentages have a variance of plus or minus percent.

## Command Output

The **show ipsec card-utilization** command displays the following information:

| Output field | Description |
|---|---|
| IPSEC Module | The module ID of the system where the ICX7400-SERVICE-MOD module is installed. |
| admin | The administration status of the ICX7400-SERVICE-MOD module. Possible values are UP or DOWN. |
| Tx pkt count | The total number of packets transmitted over the ICX7400-SERVICE-MOD module. |
| Rx pkt count | The total number of packets received from the ICX7400-SERVICE-MOD module. |
| Tx pkt/sec | The packet transmission rate over the ICX7400-SERVICE-MOD module. |
| Rx pkt/sec | The transmission rate of packets received from the ICX7400-SERVICE-MOD module. |
| Tx byte count | The total number of bytes transmitted over the ICX7400-SERVICE-MOD module. |
| Rx byte count | The total number of bytes received from the ICX7400-SERVICE-MOD module. |
| Tx bytes/sec | The packet transmission rate (in bytes) over the ICX7400-SERVICE-MOD module. |
| Rx bytes/sec | The transmission rate (in bytes) of packets received from the ICX7400-SERVICE-MOD module. |
| Encrypt In Utilization | Plain text packet received by the router for encryption. |
| Decrypt In Utilization | Encrypted packet received by the router for decryption. |
| Encrypt Out Utilization | Encrypted packet going out of the router. |
| Decrypt Out Utilization | Plain text packet going out of the router after decryption. |

# Examples

The following example shows how to display information about utilization of the ICX7400-SERVICE-MOD interface module when the maximum amount of traffic is ingressing on the device.

```
device# show ipsec card-utilization

IPSEC Module     : 1/4, admin: UP

card-utilization :
 Tx pkt count           : 2181783416   Rx pkt Count            : 2181782549
 Tx pkt/sec             : 30104535     Rx pkt/sec              : 30104483
 Tx byte count          : 208735473958 Rx byte Count           : 306526015230
 Tx bytes/sec           : 4166829344   Rx bytes/sec            : 4890915108
 Encrypt In Utilization : 53.75%       Encrypt Out Utilization : 100.00%
 Decrypt In Utilization : 100.00%      Decrypt Out Utilization : 93.24%
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ipsec profile

Displays configuration information about IP security (IPsec) profiles.

## Syntax

**show ipsec profile** [ *profile-name* ]

## Parameters

*profile-name*

Specifies the name of an IPsec profile.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When an IPsec profile is not specified, this command displays configuration information for all IPsec profiles.

## Command Output

The **show ipsec profile** command displays the following information:

| Output field | Description |
|---|---|
| Name | The name of an IPsec profile. |
| Description | A description of the IPsec profile. |
| Ike Profile | The name of the IKEv2 profile that is attached to this IPsec profile. |
| Lifetime | The lifetime period (in minutes) for an IPsec SA. The range is from 10 through 1440. The default value is 480 minutes (8 hours). A value of 0 indicates that the IPsec SA remains up indefinitely. |
| Anti-replay service | |
| DH group | The Diffie-Hellman group that is used for IKEv2 negotiations. |
| Proposal | The name of any IPsec proposals that are attached to this IPsec profile. |

# Examples

The following example shows how to display IPsec profile configuration information.

```
device# show ipsec profile
================================================================================
Name                 : 17
Description          : 17
Ike Profile          : 17
Lifetime             : 28800 sec
Anti-Replay Service  : Disabled
DH Group             : None
Proposal             : 17
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ipsec proposal

Displays configuration information about IP security (IPsec) proposals.

## Syntax

**show ipsec proposal** [ *proposal-name* ]

## Parameters

*proposal-name*

Specifies the name of an IPsec proposal.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

When an IPsec proposal is not specified, this command displays configuration information for all IPsec proposals.

## Command Output

The **show ipsec proposal** command displays the following information:

| Output field | Description |
|---|---|
| Name | The name of the IPsec proposal. |
| Protocol | The transform type. |
| Encryption | A list of encryption algorithms that are supported. |
| Authentication | The authentication method for data traffic. |
| ESN | The Extended Sequence Number (ESN) status. |
| Mode | The packet encapsulation mode that is supported. |
| Ref Count | The number of IPsec profiles that refer to this IPsec proposal. |

# Examples

The following example shows how to display configuration information for all IPsec proposals. In this example, only the default proposal (**def-ipsec-prop**) is configured on the device.

```
device# show ipsec proposal
================================================================================
Name                : def-ipsec-prop
Protocol            : ESP
Encryption          : aes-gcm-256
Authentication      : NULL
ESN                 : Disable
Mode                : Tunnel
Ref Count           : 1
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show ipsec sa

Displays configuration information about current IP security (IPsec) security associations (SAs).

## Syntax

**show ipsec sa**

**show ipsec sa address** { *ip-address* | *ipv6-address* } [ **detail** ]

**show ipsec sa detail**

**show ipsec sa identity address** { *ip-address* | *ipv6-address* }

**show ipsec sa identity dn** *dn-name*

**show ipsec sa identity email** *email-address*

**show ipsec sa identity fqdn** *fqdn-name*

**show ipsec sa identity key-id** *key-id*

**show ipsec sa interface** *tunnel-port* [ **detail** ]

**show ipsec sa ipv4**

**show ipsec sa ipv6**

**show ipsec sa peer** { *ip-address* | *ipv6-address* } [ **detail** ]

## Parameters

*ip-address*
Specifies the IPv4 address of the SA.

*ipv6-address*
Specifies the IPv6 address of the SA.

**detail**
Specifies detailed information.

**identity**
Specifies the remote identity of the SA.

**dn** *dn-name*
Specifies a Distinguished Name (DN).

**email** *email-address*
Specifies an email address.

**fqdn** *fqdn-name*
Specifies a fully qualified domain name (FQDN).

**key-id** *key-id*
Specifies a key ID.

**interface** *tunnel-port*
Specifies a tunnel port number.

**ipv4**

Specifies the IPv4 IPsec SA database.

**ipv6**

Specifies the IPv6 IPsec SA database.

**peer**

Specifies the peer address of the SA.

# Modes

User EXEC mode

# Usage Guidelines

This command may be entered in all configuration modes.

When the **detail** option is omitted, only the basic SA information is displayed.

# Command Output

The **show ipsec sa** command display the following information (when the **detail** option is specified).

| Output field | Description |
|---|---|
| interface | The IPsec tunnel interface ID. |
| Local address | The source address of the IPsec SA. |
| Remote address | The destination address of the IPsec SA. |
| Inner VRF | The base VRF of the IPsec tunnel interface. |
| Local Identity | The total traffic selector. |
| Remote Identity | The received traffic selector. |
| DF-bit | The "Don't fragment" bit that indicates if fragmentation is enabled or disabled. |
| Profile-name | The name of the IPsec profile that is used by this IPsec SA. |
| DH group | The Diffie-Hellman group that is used by this IPsec SA. |
| Direction | The direction of the IPsec SA. Possible values are INBOUND or OUTBOUND. |
| Mode | The encapsulation type. |
| Protocol | The transform type. |
| ICV size | The integrity check value (ICV) size. |
| lifetime(sec) | The rekey time for this IPsec SA. |
| Anti-replay service | The anti-replay service configuration. Possible values are Enable or Disable. |
| ESN | The Extended Sequence Number (ESN) configuration. Possible values are Enable or Disable. |
| Status | The state of the IPsec SA. |
| Worry Metric | The rekey time for the IKEv2 SA. |

# Examples

The following example displays basic information about the IPsec SA database.

```
device# show ipsec sa

IPSEC Security Association Database is empty.
SPDID(vrf:if) Dir Encap SPI        Destination   AuthAlg  EncryptAlg
IPSEC Security Association Database(child SA pair:4)
0:tnl 18    OUT IPSEC_ 0x00007935 10.18.3.4      Null     aes-gcm-256
0:tnl 18    IN  IPSEC_ 0x0000b278 10.18.3.5      Null     aes-gcm-256
0:tnl 22    OUT IPSEC_ 0x000064b2 10.22.3.4      Null     aes-gcm-256
0:tnl 22    IN  IPSEC_ 0x00008dea 10.22.3.5      Null     aes-gcm-256
0:tnl 19    OUT IPSEC_ 0x00006018 10.19.3.4      Null     aes-gcm-256
0:tnl 19    IN  IPSEC_ 0x000062df 10.19.3.5      Null     aes-gcm-256
0:tnl 20    OUT IPSEC_ 0x0000de58 10.20.3.4      Null     aes-gcm-256
0:tnl 20    IN  IPSEC_ 0x0000acff 10.20.3.5      Null     aes-gcm-256
```

The following example displays detailed information for an IPsec SA by specifying the local IP address of the SA.

```
device# show ipsec sa address 10.19.3.4 detail

IPSEC Security Association Database(child SA pair:0)
    interface           : tnl 19
    Local address: 10.3.3.4/500, Remote address: 10.19.3.5/500
    Inner VRF           : vrf1
     Local Identity (addr/mask/prot/port): address(0.0.0.0/0/0/0)
     Remote Identity(addr/mask/prot/port): address(0.0.0.0/0/0/0)
    DF-bit              : clear
    Profile-name        : 19
    DH group            : none
    Direction           : outbound, SPI: 0x6018
    Mode                : tunnel,
    Protocol            : IPSEC_ESP , Encryption : aes-gcm-256 , Authentication : Null
    ICV size            : 16 bytes
    lifetime(sec)       : Expiring in 243 secs
    Anti-replay service : Disable
    ESN                 : Disable
    Status              : ACTIVE
    Worry Metric        :0
```

The following example displays IPsec SA information, including information about IPv6 connections.

```
device# show ipsec sa

IPSEC Security Association Database(child SA pair:7)

SPDID(vrf:if) Dir   Encap        SPI    Destination       AuthAlg   EncryptAlg
  0:tnl 8     OUT IPSEC_ESP  0x000056c1 2220::1           NULL      aes-gcm-256
  0:tnl 8     IN  IPSEC_ESP  0x00004b95 5002::2           NULL      aes-gcm-256
  0:tnl 7     OUT IPSEC_ESP  0x00001489 1110::1           NULL      aes-gcm-256
  0:tnl 7     IN  IPSEC_ESP  0x000000a3 5002::2           NULL      aes-gcm-256
  0:tnl 1     OUT IPSEC_ESP  0x0000e1c1 1000::1           NULL      aes-gcm-256
  0:tnl 1     IN  IPSEC_ESP  0x00007eb2 1004::2           NULL      aes-gcm-256
  0:tnl 4     OUT IPSEC_ESP  0x00001044 120.1.1.          NULL      aes-gcm-256
  0:tnl 4     IN  IPSEC_ESP  0x00009dd5 110.1.1.1         NULL      aes-gcm-256
  0:tnl 11    OUT IPSEC_ESP  0x00000682 1000::1           NULL      aes-gcm-256
  0:tnl 11    IN  IPSEC_ESP  0x00001c49 1003::2           NULL      aes-gcm-256
  0:tnl 9     OUT IPSEC_ESP  0x0000f369 3330::1           NULL      aes-gcm-256
  0:tnl 9     IN  IPSEC_ESP  0x00005f22 5002::2           NULL      aes-gcm-256
  0:tnl 3     OUT IPSEC_ESP  0x0000f948 100.1.1.1         NULL      aes-gcm-256
  0:tnl 3     IN  IPSEC_ESP  0x000043dc 104.1.1.2         NULL      aes-gcm-256
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support was added for IPv6. |

# show ipv6

Displays the details of the IPv6 configuration.

## Syntax

**show ipv6**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show ipv6** command.

```
device# show ipv6
DUT(config-if-mgmt-1)#sh ipv6
Global Settings
  IPv6 is enabled
  Link-local address(es):
    fe80::768e:f8ff:fef9:6d80 [Preferred]
  Global unicast address(es):
    2620:100:c:fe23:768e:f8ff:fef9:6d80 [Preferred],  subnet is 2620:100:c:fe23::/64
  Joined group address(es):
    ff02::1:fff9:6d80
    ff02::1
  Best Default Router : 2620:100:c:fe23:10:37:65:129 PMTUS : 0
  MTU is 1500 bytes
  ND DAD is enabled, number of DAD attempts: 3
  ND reachable time is 30000 miliseconds
  ND retransmit interval is 1000 miliseconds
  Current Hop Limit is 64
  Hosts use stateless autoconfig for addresses
  No Inbound Access List Set
  No Outbound Access List Set
  No IPv6 Domain Name Set
  No IPv6 DNS Server Address set
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | The output was updated to display the best default router for the default gateway. |

# show ipv6 access-list

Displays the IPv6 access control lists (ACLs) configured on a device.

## Syntax

**show ipv6 access-list** [ *acl-name* ]

## Parameters

*acl-name*

Specifies the IPv6 ACL name.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

ACL configuration mode

## Usage Guidelines

From FastIron release 08.0.50, sequence numbers are automatically added to existing ACL rules, in the following manner:

- The first rule within each ACL is numbered 10.
- The sequence number for each succeeding rule is incremented by 10.

# Examples

The following example displays information about all the IPv6 ACLs configured.

```
device# show ipv6 access-list
ipv6 access-list v6-ACL1: 1 entries
10: deny ipv6 any any

ipv6 access-list v6-ACL2: 1 entries
10: permit ipv6 any any

ipv6 access-list v6-ACL3: 2 entries
10: deny ipv6 2001:DB8:10::/64 any
20: permit ipv6 any any

ipv6 access-list v6-ACL4: 2 entries
10: deny ipv6 2001:DB8::/64 any
20: permit ipv6 any any

ipv6 access-list rate-ACL: 1 entries
10: permit ipv6 any any traffic-policy rate800M

ipv6 access-list v6-ACL5: 8 entries
10: permit tcp 2001:DB8::/64 any
20: permit ipv6 2001:DB8::/64 any
30: permit ipv6 2001:DB8:101::/64 any
40: permit ipv6 2001:DB8:10::/64 2001:DB8:102::/64
50: permit ipv6 host 2001:DB8:aa:10::102 host 2001:DB8:101::102
60: permit ipv6 host 2001:DB8:10::101 host 2001:DB8:101::101 dscp-matching 0
70: dscp-marking 63 dscp-cos-mapping
80: permit ipv6 any any dscp-matching 63 dscp-cos-mapping
90: permit ipv6 any any fragments
```

The following example displays information for a specific IPv6 ACL.

```
device# show ipv6 access-list rtr
ipv6 access-list rtr: 3 entries
10: remark This entry permits ipv6 packets from 2001:DB8::2 to any destination permit ipv6 host
2001:DB8::2 any
20: remark This entry denies udp packets from any source to any destination deny udp any any
30: remark This entry denies IPv6 packets from any source to any destination deny ipv6 any any
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | The command was modified to add sequence numbers automatically to existing rules. |

# show ipv6 bgp

Displays entries in the BGP4+ routing table.

## Syntax

**show ipv6 bgp**

**show ipv6 bgp** *ipv6-prefix /prefix-length*

**show ipv6 bgp** *ipv6-prefix /prefix-length* **longer-prefixes**

## Parameters

*ipv6-prefix*

Specifies an IPv6 network number.

*/prefix-length*

Specifies the length of the IPv6 prefix.

**longer-prefixes**

Displays routes that match a specified or longer BGP prefix.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp** command displays the following information:

| Output field | Description |
|---|---|
| Total number of BGP Routes (appears in display of all BGP routes only) | The number of routes known by the device. |
| Number of BGP Routes matching display condition (appears in display that matches specified and longer prefixes) | The number of routes that matched the display parameters you entered. This is the number of routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command's output. |
| Origin codes | A character the display uses to indicate the route's origin. The origin code appears to the right of the AS path (Path field). The origin codes are described in the command's output. |
| Network | The network prefix and prefix length. |
| Next Hop | The next-hop router for reaching the network from the device. |
| MED | The value of the route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for this route relative to other routes in the local AS. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295. |

| Output field | Description |
|---|---|
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Path | The route's AS path. |

# Examples

The following example displays sample output from the **show ipv6 bgp** command.

```
device> show ipv6 bgp

Total number of BGP Routes: 4
Status codes: s suppressed, d damped, h history, * valid, > best, i internal, S
stale
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network                   Next Hop     MED    LocPrf     Weight Path
*>  2001:db8:10:10::/64 ::              1      100        32768  ?
*>  2001:db8:113:113::/64  ::
                                        1      100        32768  i
*>  2001:db8:400:400::/64  ::1
                                        0      100        32768  i
*i  2001:db8:400:400::/64  2001:db8:400:400::2
                                        0      400        0      65005 65010 ?
*>i 2001:db8:824:824::/64  2001:db8:400:400::2
                                        0      400        0      65005 65010 i
```

The following example displays sample output from the **show ipv6 bgp** command, showing information for prefix 2001:db8:400:400::/64, when the **longer-prefixes** keyword is used.

```
device> show ipv6 bgp 2001:db8:400:400::/64 longer-prefixes

Number of BGP Routes matching display condition : 2
Status codes: s suppressed, d damped, h history, * valid, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop        MED    LocPrf     Weight Path
*>  2001:db8:400:400::/64  ::
                                       0      100        32768  i
*i  2001:db8:400:400::/64  2001:db8:400:400::2
                                       0      400        0      65005 65010 ?
```

# show ipv6 bgp attribute-entries

Displays BGP4+ route-attribute entries that are stored in device memory.

## Syntax

**show ipv6 bgp attribute-entries**

## Modes

User EXEC mode

## Usage Guidelines

The route-attribute entries table lists the sets of BGP4+ attributes that are stored in device memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the device typically has fewer attribute entries than routes. Use this command to view BGP4+ route-attribute entries that are stored in device memory.

## Command Output

The **show ipv6 bgp attribute-entries** command displays the following information:

| Output field | Description |
|---|---|
| Total number of BGP Attribute Entries | The number of entries contained in the device's BGP4+ route-attribute entries table. |
| Next Hop | The IPv6 address of the next hop router for routes that have this set of attributes. |
| MED | The cost of the routes that have this set of attributes. |
| Origin | The source of the route information. The origin can be one of the following:<br>• EGP - The routes with this set of attributes came to BGP4+ through EGP.<br>• IGP - The routes with this set of attributes came to BGP4+ through IGP.<br>• INCOMPLETE - The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPFv3 or RIPng.<br>When BGP4+ compares multiple routes to a destination to select the best route, IGP is preferred over EGP, and both are preferred over INCOMPLETE. |
| Originator | The originator of the route in a route-reflector environment. |
| Cluster List | The route-reflector clusters through which this set of attributes has passed. |
| Aggregator | Aggregator information:<br>• AS Number shows the AS in which the network information in the attribute set was aggregated. This value applies only to aggregated routes and is otherwise 0.<br>• Router-ID shows the router that originated this aggregator. |
| Atomic | Whether the network information in this set of attributes has been aggregated and this aggregation has resulted in information loss:<br>• TRUE - Indicates information loss has occurred<br>• FALSE - Indicates no information loss has occurred<br>• None - Indicates this attribute is not present. |

| Output field | Description |
|---|---|
| | **NOTE**<br>Information loss under these circumstances is a normal part of BGP4+ and does not indicate an error. |
| Local Pref | The degree of preference for routes that use this set of attributes relative to other routes in the local AS. |
| Communities | The communities that routes with this set of attributes are in. |
| AS Path | The ASs through which routes with this set of attributes have passed. The local AS is shown in parentheses. |
| Address | For debugging purposes only. |
| Hash | For debugging purposes only. |
| Links | For debugging purposes only. |
| Reference Counts | For debugging purposes only. |

# Examples

The following example show sample output for the **show ip bgp attribute-entries** command.

```
device> show ipv6 bgp attribute-entries

Total number of BGP Attribute Entries: 4
1       Next Hop   : ::                                          MED :1
        Origin:IGP
        Originator:0.0.0.0          Cluster List:None
        Aggregator:AS Number :0     Router-ID:0.0.0.0          Atomic:None
        Local Pref:100              Communities:Internet
        AS Path    : (length 0)
        AsPathLen: 0  AsNum: 0,     SegmentNum: 0, Neighboring As: 1, Source As 0
        Address: 0x2a8bd092  Hash:364 (0x1000000)
        Links: 0x0, 0x0
        Reference Counts: 2:0:4, Magic: 3
…
```

# show ipv6 bgp config

Displays active BGP4+ configuration information.

## Syntax

**show ipv6 bgp config**

## Modes

User EXEC mode

## Examples

The following example displays the active BGP4+ configuration information contained in the running configuration without displaying the entire running configuration.

```
device> show ipv6 bgp config

Current BGP configuration:
router bgp
local-as 65020
default-local-preference 400
neighbor 8.8.8.2 remote-as 65080
neighbor 140.140.140.1 remote-as 65020
neighbor 2001:db8:400:400::3 remote-as 65020
neighbor 2001:db8:400:400::3 soft-reconfiguration inbound
address-family ipv6 unicast
neighbor 2001:db8:400:400::3 activate
neighbor 2001:db8:400:400::3 route-map in bgp_map
exit-address-family
end
```

# show ipv6 bgp dampened-paths

Displays all BGP4+ dampened routes.

## Syntax

**show ipv6 bgp dampened-paths**

## Modes

User EXEC mode

## Command Output

The **show ip bgp dampened-paths** command displays the following information:

| Output field | Description |
|---|---|
| Status codes | A list of the characters the display uses to indicate the path's status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command's output. The status column displays a "d" for each dampened route. |
| Network | The destination network of the route. |
| From | The IPv6 address of the advertising peer. |
| Flaps | The number of times the path has flapped. |
| Since | The amount of time (in hh:mm:ss) since the first flap of this route. |
| Reuse | The amount of time (in hh:mm:ss) after which the path is available again. |
| Path | The AS path of the route. |

## Examples

The following example displays BGP4+ paths that have been dampened (suppressed) by route flap dampening.

```
device> show ipv6 bgp dampened-paths

Status Code >:best d:damped h:history *:valid
    Network              From                  Flaps     Since     Reuse     Path
*d     2001:db8::/13        2001:db8:1::1     1     0 :1 :14     0 :2 :20     100 1002 1000
*d     2001:db8::/16              2001:db8:1::1     1     0 :1 :14     0 :2 :20     100 1002 1000
*d     2001:db8::/14              2001:db8:1::1     1     0 :1 :14     0 :2 :20     100 1002 1000
*d     2001:db8::/15              2000:1:1::1       1  0 :1 :14     0 :2 :20     100 1002 1000
*d     2001:db8:8000::/17 2001:db8:1::1 1     0 :1 :14     0 :2 :20     100 1002 1000
*d     2001:db8:1:17::/64    2001:db8:1::1 1     0 :1 :18     0 :2 :20     100
```

# show ipv6 bgp filtered-routes

Displays BGP4+ filtered routes that are received from a neighbor or peer group.

## Syntax

**show ipv6 bgp filtered-routes** [ **detail** ] [ *ipv6-addr* { **/** *mask* } [ **longer-prefixes** ] | **as-path-access-list** *name* | **prefix-list** *name* ]

## Parameters

**detail**

Displays detailed route information.

*ipv6-addr*

Specifies the IPv6 address of the destination network in dotted-decimal notation.

*mask*

Specifies the IPv6 mask of the destination network in CIDR notation.

**longer-prefixes**

Specifies all statistics for routes that match the specified route, or that have a longer prefix than the specified route.

**as-path-access-list** *name*

Specifies an AS-path ACL. The name must be between 1 and 32 ASCII characters in length.

**prefix-list** *name*

Specifies an IPv6 prefix list. The name must be between 1 and 32 ASCII characters in length.

*name*

Specifies the name of an AS-path ACL or prefix list.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp filtered-routes** command displays the following information:

| Output field | Description |
|---|---|
| Number of BGP4+ Routes matching display condition | The number of routes that matched the display parameters you entered. This is the number of routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command's output. The status column displays an "IF" for each filtered route. |
| Prefix | The network address and prefix. |
| Next Hop | The next-hop router for reaching the network from the device. |

| Output field | Description |
|---|---|
| MED | The value of the route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for this route relative to other routes in the local AS. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295. |
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Status | The route's status, which can be one or more of the following:<br>• A - AGGREGATE - The route is an aggregate route for multiple networks.<br>• B - BEST - BGP4+ has determined that this is the optimal route to the destination.<br>• b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, RIPng, or static IPv6 routes).<br>• C - CONFED_EBGP - The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation.<br>• D - DAMPED - This route has been dampened (by the route dampening feature), and is currently unusable.<br>• E - EBGP - The route was learned through a in another AS.<br>• H - HISTORY - Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.<br>• I - IBGP - The route was learned through a in the same AS.<br>• L - LOCAL - The route originated on this device.<br>• M - MULTIPATH - BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".<br>**NOTE**<br>If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.<br>• S - SUPPRESSED - This route was suppressed during aggregation and thus is not advertised to neighbors.<br>• F - FILTERED - This route was filtered out by BGP4+ route policies on the device, but the device saved updates containing the filtered routes. |

# Examples

The following example displays BGP4+ filtered routes.

```
device> show ipv6 bgp filtered-routes

Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix                                    Next Hop                              MED    LocPrf
Weight Status
1      2001:db8:2:2::/64    2001:db8:400:400::3         0                 100             0        IF
             AS_PATH:

2      2001:db8:10:10::/64     2001:db8:400:400::3         0                 100             0        IF
             AS_PATH:

         AS_PATH:
```

The following example displays detailed information for BGP4+ filtered routes.

```
device> show ipv6 bgp filtered-routes detail

Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
1     Prefix: 2001:db8:1::/64, Status: EF, Age: 0h0m10s
    NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH: 100
2     Prefix: 2001:db8:18::/64, Status: EF, Age: 0h0m10s
    NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH: 100
3     Prefix: 2001:db8:1::/64, Status: EF, Age: 0h0m10s
    NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH: 100
4     Prefix: 2001:db8:1::/64, Status: EF, Age: 0h0m10s
    NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH: 100
5     Prefix: 2001:db8:11::1/128, Status: EF, Age: 0h0m10s
    NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: igp, Weight: 0
    AS_PATH: 100
6     Prefix: 2001:db8:17::/64, Status: EF, Age: 0h0m10s
    NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
    LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
    AS_PATH: 100
```

# show ipv6 bgp flap-statistics

Displays BGP4+ route-dampening statistics for all dampened routes with a variety of options.

## Syntax

**show ipv6 bgp flap-statistics**

**show ipv6 bgp flap-statistics** *ipv6-addr* { **/** *mask* } [ **longer-prefix** ]

**show ipv6 bgp flap-statistics as-path-filter** *name*

**show ipv6 bgp flap-statistics neighbor** *ipv6-addr*

**show ipv6 bgp flap-statistics regular-expression** *name*

## Parameters

*ipv6-addr*
IPv6 address of a specified route in dotted-decimal notation.

*mask*
IPv6 mask of a specified route in CIDR notation.

**longer-prefixes**
Displays statistics for routes that match the specified route or have a longer prefix than the specified route.

**as-path-filter** *name*
Specifies an AS-path filter.

**neighbor**
Displays flap statistics only for routes learned from the specified neighbor.

*ip-addr*
IPv4 address of the neighbor.

**regular-expression**
Specifies a regular expression in the display output on which to filter.

*name*
Name of an AS-path filter or regular expression.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp flap-statistics** command displays the following information:

| Output field | Description |
|---|---|
| Total number of flapping routes | The total number of routes in the device's BGP4+ route table that have changed state and thus have been marked as flapping routes. |

| Output field | Description |
|---|---|
| Status code | Indicates the dampening status of the route, which can be one of the following:<br>• > - This is the best route among those in the BGP4+ route table to the route's destination.<br>• d - This route is currently dampened, and thus unusable.<br>• h - The route has a history of flapping and is unreachable now.<br>• * - The route has a history of flapping but is currently usable. |
| Network | The destination network of the route. |
| From | The IPv6 address of the advertising peer. |
| Flaps | The number of flaps (state changes) the route has experienced. |
| Since | The amount of time (in hh:mm:ss) since the first flap of this route. |
| Reuse | The amount of time (in hh:mm:ss) after which the path is again available. |
| Path | The AS path of the route. |

# Examples

The following example displays route dampening statistics.

```
device> show ipv6 bgp flap-statistics

Total number of flapping routes: 14
    Status Code  >:best d:damped h:history *:valid
    Network          From           Flaps  Since    Reuse    Path
h>  2001:db8::/32    2001:db8::47   1      0 :0 :13 0 :0 :0  65001 4355 1 701
*>  2001:db8::/32    2001:db8::47   1      0 :1 :4  0 :0 :0  65001 4355 701 62
```

# show ipv6 bgp neighbors

Displays configuration information and statistics for BGP4+ neighbors of the device.

## Syntax

**show ipv6 bgp neighbors**

**show ipv6 bgp neighbors** *ipv6-addr*

**show ipv6 bgp neighbors last-packet-with-error**

**show ipv6 bgp neighbors routes-summary**

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**

Displays information about the last packet from a neighbor that contained an error.

**routes-summary**

Displays information about all route information received in UPDATE messages from BGP neighbors.

## Modes

User EXEC mode

## Usage Guidelines

Use this command to view configuration information and statistics for BGP neighbors of the device. Output shows all configured parameters for the neighbors. Only the parameters whose values differ from defaults are shown.

## Command Output

The **show ipv6 bgp neighbors** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IPv6 address of the neighbor. |
| AS | The AS in which the neighbor resides. |
| EBGP or IBGP | Whether the neighbor session is an IBGP session, an EBGP session, or a confederation EBGP session:<br>• EBGP - The neighbor is in another AS.<br>• EBGP_Confed - The neighbor is a member of another sub-AS in the same confederation.<br>• IBGP - The neighbor is in the same AS. |
| RouterID | The neighbor's router ID. |

| Output field | Description |
|---|---|
| State | The state of the device's session with the neighbor. The states are from the perspective of the session, not the neighbor's perspective. The state values can be one of the following:<br>• IDLE - The BGP4+ process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4+ process.<br>– A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• ADMND - The neighbor has been administratively shut down.<br>– A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• CONNECT - BGP4+ is waiting for the connection process for the TCP neighbor session to be completed.<br>• ACTIVE - BGP4+ is waiting for a TCP connection from the neighbor.<br>**NOTE**<br>If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.<br>• OPEN SENT - BGP4+ is waiting for an Open message from the neighbor.<br>• OPEN CONFIRM - BGP4+4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the device receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.<br>• ESTABLISHED - BGP4+ is ready to exchange UPDATE messages with the neighbor.<br>– If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed.<br>**NOTE**<br>If you display information for the neighbor using the **show ipv6 bgp neighbor**<*ipv6-address*> command, the TCP receiver queue value will be greater than 0. |
| Time | The amount of time this session has been in its current state. |
| KeepAliveTime | The keep alive time, which specifies how often this device sends keep alive messages to the neighbor. |
| HoldTime | The hold time, which specifies how many seconds the device will wait for a KEEPALIVE or UPDATE message from a BGP4+ neighbor before deciding that the neighbor is dead. |
| RefreshCapability | Whether the device has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability. |
| Messages Sent and Received | The number of messages this device has sent to and received from the neighbor. The display shows statistics for the following message types:<br>• Open<br>• Update<br>• KeepAlive<br>• Notification<br>• Refresh-Req |
| Last Update Time | Lists the last time updates were sent and received for the following:<br>• NLRIs<br>• Withdraws |
| Last Connection Reset Reason | The reason the previous session with this neighbor ended. The reason can be one of the following:<br>• No abnormal error has occurred. |

| Output field | Description |
|---|---|
| | • Reasons described in the BGP specifications:<br>– Message Header Error<br>– Connection Not Synchronized<br>– Bad Message Length<br>– Bad Message Type<br>– OPEN Message Error<br>– Unsupported Version Number<br>– Bad Peer AS Number<br>– Bad BGP Identifier<br>– Unsupported Optional Parameter<br>– Authentication Failure<br>– Unacceptable Hold Time<br>– Unsupported Capability<br>– UPDATE Message Error<br>– Malformed Attribute List<br>– Unrecognized Well-known Attribute<br>– Missing Well-known Attribute<br>– Attribute Flags Error<br>– Attribute Length Error<br>– Invalid ORIGIN Attribute<br>– Invalid NEXT_HOP Attribute<br>– Optional Attribute Error<br>– Invalid Network Field<br>– Malformed AS_PATH<br>– Hold Timer Expired<br>– Finite State Machine Error<br>– Rcv Notification |
| Last Connection Reset Reason (cont.) | • Reasons specific to the implementation:<br>– Reset All Peer Sessions<br>– User Reset Peer Session<br>– Port State Down<br>– Peer Removed<br>– Peer Shutdown<br>– Peer AS Number Change<br>– Peer AS Confederation Change<br>– TCP Connection KeepAlive Timeout<br>– TCP Connection Closed by Remote<br>– TCP Data Stream Error Detected |
| Notification Sent | If the device receives a NOTIFICATION message from the neighbor, the message contains an error code corresponding to one of the following errors. Some errors have subcodes that clarify the reason for the error. Where applicable, the subcode messages are listed underneath the error code messages.<br>• Message Header Error<br>– Connection Not Synchronized<br>– Bad Message Length<br>– Bad Message Type<br>– Unspecified<br>• Open Message Error<br>– Unsupported Version<br>– Bad Peer As<br>– Bad BGP Identifier<br>– Unsupported Optional Parameter<br>– Authentication Failure<br>– Unacceptable Hold Time<br>– Unspecified<br>• Update Message Error<br>– Malformed Attribute List |

| Output field | Description |
|---|---|
| | – Unrecognized Attribute<br>– Missing Attribute<br>– Attribute Flag Error<br>– Attribute Length Error<br>– Invalid Origin Attribute<br>– Invalid NextHop Attribute<br>– Optional Attribute Error<br>– Invalid Network Field<br>– Malformed AS Path<br>– Unspecified<br>• Hold Timer Expired<br>• Finite State Machine Error<br>• Cease<br>• Unspecified |
| Notification Received | See above. |
| Neighbor NLRI Negotiation | The state of the device's NLRI negotiation with the neighbor. The states can include the following:<br>• Peer negotiated IPv6 unicast capability.<br>• Peer configured for IPv6 unicast routes.<br>• Peer negotiated IPv4 unicast capability.<br>• Peer negotiated IPv4 multicast capability. |
| TCP Connection state | The state of the connection with the neighbor. The connection can have one of the following states:<br>• LISTEN - Waiting for a connection request.<br>• SYN-SENT - Waiting for a matching connection request after having sent a connection request.<br>• SYN-RECEIVED - Waiting for a confirming connection request acknowledgment after having both received and sent a connection request.<br>• ESTABLISHED - Data can be sent and received over the connection. This is the normal operational state of the connection.<br>• FIN-WAIT-1 - Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent.<br>• FIN-WAIT-2 - Waiting for a connection termination request from the remote TCP.<br>• CLOSE-WAIT - Waiting for a connection termination request from the local user.<br>• CLOSING - Waiting for a connection termination request acknowledgment from the remote TCP.<br>• LAST-ACK - Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request).<br>• TIME-WAIT - Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request.<br>• CLOSED - There is no connection state. |
| Byte Sent | The number of bytes sent. |
| Byte Received | The number of bytes received. |
| Local host | The IPv6 address of the device. |
| Local port | The TCP port the Ruckus device is using for the BGP4+ TCP session with the neighbor. |
| Remote host | The IPv6 address of the neighbor. |
| Remote port | The TCP port the neighbor is using for the BGP4+ TCP session with the device. |
| ISentSeq | The initial send sequence number for the session. |
| SendNext | The next sequence number to be sent. |

| Output field | Description |
|---|---|
| TotUnAck | The number of sequence numbers sent by the device that have not been acknowledged by the neighbor. |
| TotSent | The number of sequence numbers sent to the neighbor. |
| ReTrans | The number of sequence numbers that the device retransmitted because they were not acknowledged. |
| UnAckSeq | The current acknowledged sequence number. |
| IRcvSeq | The initial receive sequence number for the session. |
| RcvNext | The next sequence number expected from the neighbor. |
| SendWnd | The size of the send window. |
| TotalRcv | The number of sequence numbers received from the neighbor. |
| DupliRcv | The number of duplicate sequence numbers received from the neighbor. |
| RcvWnd | The size of the receive window. |
| SendQue | The number of sequence numbers in the send queue. |
| RcvQue | The number of sequence numbers in the receive queue. |
| CngstWnd | The number of times the window has changed. |

# Examples

The following is sample output from the **show ipv6 bgp neighbors** command when no arguments or keywords are used.

```
device> show ipv6 bgp neighbors

    Total number of BGP Neighbors: 2
1   IP Address: 2001:1001::1, AS: 63753 (IBGP), RouterID: 1.0.0.1, VRF: default-vrf
       Description: SWD-2
    State: ESTABLISHED, Time: 0h47m50s, KeepAliveTime: 60, HoldTime: 180
       KeepAliveTimer Expire in 26 seconds, HoldTimer Expire in 168 seconds
    Minimal Route Advertisement Interval: 0 seconds
       MD5 Password: $Qj0tZHMlXC1vbjYt
       UpdateSource: Loopback 1
       NextHopSelf: yes
       RefreshCapability: Received
       GracefulRestartCapability: Received
           Restart Time 120 sec, Restart bit 0
           afi/safi 2/1, Forwarding bit 0
       GracefulRestartCapability: Sent
           Restart Time 120 sec, Restart bit 0
           afi/safi 2/1, Forwarding bit 0
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
.....
```

The following is sample output from the **show ipv6 bgp neighbors** command when an IPv6 address is specified.

```
device> show ipv6 bgp neighbors 2001:db8:113:113::2

Total number of BGP Neighbors: 2
1   IP Address: 2001:db8:113:113::2, AS: 65001 (EBGP), RouterID: 0.0.0.0, VRF:
efault-vrf
    State: CONNECT, Time: 1d14h21m38s, KeepAliveTime: 60, HoldTime: 180
    Minimal Route Advertisement Interval: 0 seconds
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 1       0       0         1            0
       Received: 1       0       0         0            0
    Last Connection Reset Reason:Unknown
    Notification Sent:     Unspecified
    Notification Received: Unspecified
    Neighbor NLRI Negotiation:
      Peer configured for IPV6 unicast  Routes
    Neighbor AS4 Capability Negotiation:
    Outbound Policy Group:
       ID: 2, Use Count: 3
       Last update time was 123948 sec ago
    TCP Connection state: SYN-SENT
     Maximum segment size: 1440
    TTL check: value: 0
       Byte Sent:   0, Received: 0
       Local host: 2001:db8:113:113::1, Local  Port: 8014
       Remote host: 2001:db8:113:113::2, Remote Port: 179
       ISentSeq:   76022806  SendNext:   76022807  TotUnAck:          1
       TotSent:           1  ReTrans:           2  UnAckSeq:   76022806
       IRcvSeq:           0  RcvNext:           0  SendWnd:           1
       TotalRcv:          0  DupliRcv:          0  RcvWnd:        16384
       SendQue:           1  RcvQue:            0  CngstWnd:       1440
…
```

# show ipv6 bgp neighbors advertised-routes

Displays the routes that the device has advertised to the neighbor during the current BGP4+ session.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **advertised-routes** [ **detail** | **/** *mask-bits* ]

## Parameters

*ipv6-addr*

Specifies the IPv6 address of a neighbor in dotted-decimal notation.

**detail**

Specifies detailed information.

*mask-bits*

Specifies the number of mask bits in CIDR notation.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbor advertised-routes** command displays the following information:

| Output field | Description |
|---|---|
| Number of BGP4+ Routes advertised to specified neighbor (appears only in display for all routes) | The number of routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output. |
| Prefix | The advertised route's prefix. |
| Next Hop | The next-hop for reaching the advertised route from the device. |
| MED | The value of the advertised route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for the advertised route relative to other routes in the local autonomous system. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference range is 0 - 4294967295. |
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Status | The advertised route's status, which can be one or more of the following:<br>• A - AGGREGATE. The route is an aggregate route for multiple networks.<br>• B - BEST. BGP4+ has determined that this is the optimal route to the destination.<br>• b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device |

| Output field | Description |
|---|---|
| | received better routes from other sources (such as OSPFv3, RIPng, or static IPv6 routes).<br>• E - EBGP. The route was learned through a in another AS.<br>• I - IBGP. The route was learned through a in the same AS.<br>• L - LOCAL. The route originated on this device. |
| AS-PATH | The AS-path information for the route. |

# Examples

The following example displays the routes the device has advertised to a specified neighbor.

```
device> show ipv6 bgp neighbor 2001:db8::110 advertised-routes

There are 2 routes advertised to neighbor 2001:db8::110
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
       Prefix           Next Hop   MED LocPrf   Weight Status
1      2001:db8::/32    ::         1            32768  BL
         AS_PATH:
2      2001:db8::/16    ::         1            32768  BL
         AS_PATH:
```

# show ipv6 bgp neighbors flap-statistics

Displays the route flap statistics for routes received from or sent to a BGP4+ neighbor.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **flap-statistics**

## Parameters

*ipv6-addr*

Specifies the IPv4 address of a neighbor in dotted-decimal notation.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbor flap-statistics** command displays the following information:

| Output field | Description |
|---|---|
| Total number of flapping routes | The total number of routes in the neighbor's BGP4+ route table that have changed state and thus have been marked as flapping routes. |
| Status code | Indicates the status of the route, which can be one of the following:<br>• > - This is the best route among those in the neighbor's BGP4+ route table to the route's destination.<br>• d - This route is currently dampened, and thus unusable.<br>• h - The route has a history of flapping and is unreachable now.<br>• * - The route has a history of flapping but is currently usable. |
| Network | The destination network of the route. |
| From | The IPv6 address of the advertising peer. |
| Flaps | The number of flaps (state changes) the route has experienced. |
| Since | The amount of time (in hh:mm:ss) since the first flap of this route. |
| Reuse | The amount of time (in hh:mm:ss) after which the path is again available. |
| Path | The AS path of the route. |

## Examples

The following example displays route flap dampening statistics for a specified BGP4+ neighbor.

```
device> show ipv6 bgp neighbor 2001:db8::110 flap-statistics

Total number of flapping routes: 14
    Status Code  >:best d:damped h:history *:valid
    Network            From            Flaps Since    Reuse    Path
h>  2001:db8::/32       10.90.213.77    1     0 :0 :13 0 :0 :0  65001 4355 1 701
*>  2001:db8::/32       10.90.213.77    1     0 :1 :4  0 :0 :0  65001 4355 701 62
```

# show ipv6 bgp neighbors last-packet-with-error

Displays the last packets with an error from BGP4+ neighbors of the device.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **last-packet-with-error** [ **decode** ]

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**decode**

Decodes the last packet that contained an error from any of a device's neighbors.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbor last-packet-with-error** command displays the following information:

| Output field | Description |
|---|---|
| Total number of BGP Neighbors | The total number of configured neighbors for a device. |
| Last error | The error packet's contents decoded in a human-readable format or notification that no packets with an error were received. |

# show ipv6 bgp neighbors received

Displays Outbound Route Filters (ORFs) received from BGP4+ neighbors of the device.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **received** { **extended-community** | **prefix-filter** }

## Parameters

*ipv6-addr*

Specifies the IPv6 address of a neighbor in dotted-decimal notation.

**extended-community**

Displays the results for ORFs that use the BGP Extended Community Attribute.

**prefix-filter**

Displays the results for ORFs that are prefix-based.

## Modes

User EXEC mode

## Examples

The following example displays sample output for the **show ipv6 bgp neighbors received** command when the **prefix-filter** keyword is used.

```
device> show ipv6 bgp neighbor 2001:db8::110 received prefix-filter

ip prefix-list 2001:db8::110: 4 entries
     seq 5 permit 2001:db8:3::45/16 ge 18 le 28
     seq 10 permit 2001:db8::4::88/24
     seq 15 permit 2001:db8:5::37/8 le 32
     seq 20 permit 2001:db8:6::83/16 ge 18
```

# show ipv6 bgp neighbors received-routes

Lists all route information received in route updates from BGP4+ neighbors of the device since the soft-reconfiguration feature was enabled.

## Syntax

**show ipv6 bgp neighbors** *iipv6-addr* **received-routes** [ **detail** ]

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**detail**

Displays detailed route information.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbor received-routes** command displays the following information:

| Output field | Description |
|---|---|
| Number of BGP4+ Routes received from a neighbor | The number of routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output. |
| Prefix | The received route's prefix. |
| Next Hop | The IPv6 address of the next device that is used when forwarding a packet to the received route. |
| MED | The value of the route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for the advertised route relative to other routes in the local autonomous system. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295. |
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Status | The advertised route's status, which can be one or more of the following:<br>A - AGGREGATE. The route is an aggregate route for multiple networks.<br>B - BEST. BGP4+ has determined that this is the optimal route to the destination.<br>b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, RIPng, or static IPv6 routes). |

| Output field | Description |
|---|---|
| | D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable.<br><br>E - EBGP. The route was learned through a in another AS.<br><br>H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.<br><br>I - IBGP. The route was learned through a in the same autonomous system.<br><br>L - LOCAL. The route originated on this device.<br><br>M - MULTIPATH. BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".<br><br>**NOTE**<br>If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.<br><br>S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors.<br><br>F - FILTERED. This route was filtered out by BGP4+ route policies on the device, but the saved updates containing the filtered routes. |

# Examples

The following example displays a summary of the route information received in route updates from neighbor 2001:db8::10.

```
device> show ipv6 bgp neighbor 2001:db8:400:400::2 received-route
       There are 4 received routes from neighbor 2001:db8:400:400::2
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix                Next Hop             MED LocPrf Weight  Status
1      2001:db8:202:202::/64  2001:db8:400:400::2  0   400    0       BI

         AS_PATH: 65005 65010
2      2001:db8:400:400::/64  2001:db8:400:400::2  0   400    0       I

         AS_PATH: 65005 65010
```

The following example displays output for the **show ipv6 bgp neighbor received-routes** when the **details** keyword is used.

```
device> show ipv6 bgp neighbor 2001:db8:1::1 received-routes detail

There are 4 received routes from neighbor 2001:db8:1::1
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
1 Prefix: 2001:db8:1000:1::/64, Status: BI, Age: 0h17m25s
NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
AS_PATH:
Adj_RIB_out count: 1, Admin distance 200
2 Prefix: 2001:db8:1::/64, Status: I, Age: 0h17m25s
NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
AS_PATH:
3 Prefix: 2001:db8:11::1/128, Status: BI, Age: 0h17m25s
NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
LOCAL_PREF: 100, MED: 0, ORIGIN: igp, Weight: 0
AS_PATH:
Adj_RIB_out count: 1, Admin distance 200
4 Prefix: 2001:db8:17::/64, Status: BI, Age: 0h17m25s
NEXT_HOP: 2001:db8:1::1, Learned from Peer: 2001:db8:1::1 (100)
LOCAL_PREF: 100, MED: 0, ORIGIN: incomplete, Weight: 0
AS_PATH:
Adj_RIB_out count: 1, Admin distance 200
```

# show ipv6 bgp neighbors rib-out-routes

Displays information about the current BGP4+ Routing Information Base (Adj-RIB-Out) for specific neighbors and specific destination networks.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **rib-out-routes** [ **detail** ] [*ipv6-addr* [ **/** *mask* ] ]

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**last-packet-with-error**

Displays the last packet with an error.

**routes-summary**

Displays routes received, routes accepted, number of routes advertised by peer, and so on.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbors rib-out-routes** command displays the following information:

| Output field | Description |
|---|---|
| Number of RIB_out routes for a specified neighbor (appears only in display for all RIB routes) | The number of RIB routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output. |
| Prefix | The RIB route's prefix. |
| Next Hop | The next-hop router for reaching the route from the device. |
| MED | The value of the advertised route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for the route relative to other routes in the local autonomous system. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295. |
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Status | The RIB route's status, which can be one or more of the following:<br>• A - AGGREGATE. The route is an aggregate route for multiple networks.<br>• B - BEST. BGP4+ has determined that this is the optimal route to the destination.<br>• E - EBGP. The route was learned through a in another autonomous system. |

| Output field | Description |
|---|---|
| | • I - IBGP. The route was learned through a in the same autonomous system.<br>• L - LOCAL. The route originated on this device. |
| AS-PATH | The AS-path information for the route. |

# Examples

The following example displays a summaryabout all RIB routes for neighbor 2001:db8::110.

```
device> show ipv6 bgp neighbor 2001:db8::110 rib-out-routes

       There are 2 RIB_out routes for neighbor 2001:db8::110
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
       Prefix             Next Hop    Metric     LocPrf     Weight Status
1      2001:db8::/32                 ::        1          100        32768  BL
         AS_PATH:
2      2001:db8::/16                      ::        1          100        32768  BL
         AS_PATH:
```

The following example displays detailed information about all RIB routes for neighbor 2001:db8::110.

```
device> show ipv6 bgp neighbor 2001:db8::110 rib-out-routes detail

                    There are 2 RIB_out routes for neighbor 2001:db8::110
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
1       Prefix: 2001:db8::/32,  Status: BL,  Age: 6d18h17m53s
         NEXT_HOP: ::,  Learned from Peer: Local Router
          LOCAL_PREF: 100,  MED: 1,  ORIGIN: incomplete,  Weight: 32768
           AS_PATH:
            Adj_RIB_out count: 1,  Admin distance 190
2       Prefix: 2001:db8::/16,  Status: BL,  Age: 6d18h21m8s
         NEXT_HOP: ::,  Learned from Peer: Local Router
          LOCAL_PREF: 100,  MED: 1,  ORIGIN: incomplete,  Weight: 32768
           AS_PATH:
            Adj_RIB_out count: 1,  Admin distance 190
            Adj_RIB_out count: 1,  Admin distance 190
```

# show ipv6 bgp neighbors routes

Lists a variety of route information received in UPDATE messages from BGP4+ neighbors.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **routes**

**show ipv6 bgp neighbors** *ipv6-addr* **routes** { **best** | **not-installed-best** | **unreachable** }

**show ipv6 bgp neighbors** *ipv6-addr* **routes detail** { **best** | **not-installed-best** | **unreachable** }

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

**best**

Displays routes received from the neighbor that are the best BGP4+ routes to their destination.

**not-installed-best**

Displays routes received from the neighbor that are the best BGP4+ routes to their destination but were not installed in the route table because the device received better routes from other sources.

**unreachable**

Displays routes that are unreachable because the device does not have a valid RIP, OSPF, or static route to the next hop.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbors routes** command displays the following information:

| Output field | Description |
|---|---|
| Number of accepted routes from a specified neighbor | The number of routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output. |
| Prefix | The route's prefix. |
| Next Hop | The next-hop router for reaching the route from the device. |
| MED | The value of the route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for the route relative to other routes in the local autonomous system. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295. |

| Output field | Description |
|---|---|
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Status | The route's status, which can be one or more of the following:<br>• A - AGGREGATE. The route is an aggregate route for multiple networks.<br>• B - BEST. BGP4+ has determined that this is the optimal route to the destination.<br>• C - CONFED_EBGP. The route was learned from a neighbor in the same confederation and autonomous system, but in a different sub-AS within the confederation.<br>• D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable.<br>• E - EBGP. The route was learned through a in another autonomous system.<br>• H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.<br>• I - IBGP. The route was learned through a in the same autonomous system.<br>• L - LOCAL. The route originated on this device.<br>• M - MULTIPATH. BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".<br>**NOTE**<br>If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.<br>• S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors.<br>• F - FILTERED. This route was filtered out by BGP4+ route policies on the device, but the saved updates containing the filtered routes. |
| AS-PATH | The AS-path information for the route. |

# Examples

The following example shows sample output for the **show ip bgp neighbors routes** command when the **best** keyword is used.

```
device> show ipv6 bgp neighbor 2001:db8::106 routes best

       There are 2 accepted routes from neighbor 2001:db8::106
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop           MED     LocPrf     Weight Status
1      2001:db8::/16                      2001:db8::106             1          100        0      BE
         AS_PATH: 65001
2      2001:db8::/32                            2001:db8::106             1          100        0
BE
         AS_PATH: 65001
```

The following example shows detailed sample output for the **show ip bgp neighbors routes** command when the **best** keyword is used.

```
device> show ipv6 bgp neighbor 2001:db8::106 routes detail best

       There are 2 accepted routes from neighbor 2001:db8::106
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
1       Prefix: 2001:db8::/16,  Status: BE,  Age: 18h48m56s
         NEXT_HOP: 2001:db8::106,  Learned from Peer: 2001:db8::106 (65001)
          LOCAL_PREF: 100,  MED: 1,  ORIGIN: incomplete,  Weight: 0
           AS_PATH: 65001
2       Prefix: 2001:db8::/32,  Status: BE,  Age: 18h48m56s
         NEXT_HOP: 2001:db8::106,  Learned from Peer: 2001:db8::106 (65001)
          LOCAL_PREF: 100,  MED: 1,  ORIGIN: incomplete,  Weight: 0
           AS_PATH: 65001
```

# show ipv6 bgp neighbors routes-summary

Displays route summary information for all neighbors or a specified neighbor.

## Syntax

**show ipv6 bgp neighbors** *ipv6-addr* **routes-summary**

## Parameters

*ipv6-addr*

IPv6 address of a neighbor in dotted-decimal notation.

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp neighbor routes-summary** command displays the following information:

| Output field | Description |
|---|---|
| IP Address | The IPv6 address of the neighbor |
| Routes Received | How many routes the device has received from the neighbor during the current BGP4+ session:<br>• Accepted or Installed - Indicates how many of the received routes the device accepted and installed in the BGP4+ route table.<br>• Filtered or Kept - Indicates how many routes were filtered out, but were nonetheless retained in memory for use by the soft reconfiguration feature.<br>• Filtered - Indicates how many of the received routes were filtered out. |
| Routes Selected as BEST Routes | The number of routes that the device selected as the best routes to their destinations. |
| BEST Routes not Installed in IPv6 Forwarding Table | The number of routes received from the neighbor that are the best BGP4+ routes to their destinations, but were nonetheless not installed in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, RIPng, or static IPv6 routes). |
| Unreachable Routes | The number of routes received from the neighbor that are unreachable because the device does not have a valid RIPng, OSPFv3, or static IPv6 route to the next hop. |
| History Routes | The number of routes that are down but are being retained for route flap dampening purposes. |
| NLRIs Received in Update Message | The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages:<br>• Withdraws - The number of withdrawn routes the device has received.<br>• Replacements - The number of replacement routes the device has received. |
| NLRIs Discarded due to | Indicates the number of times the device discarded an NLRI for the neighbor due to the following reasons:<br>• Maximum Prefix Limit - The device's configured maximum prefix amount had been reached.<br>• AS Loop - An AS loop occurred. An AS loop occurs when the BGP4+ AS-path attribute contains the local AS number. |

| Output field | Description |
|---|---|
| | • Invalid Nexthop Address - The next hop value was not acceptable.<br>• Duplicated Originator_ID - The originator ID was the same as the local router ID.<br>• Cluster_ID - The cluster list contained the local cluster ID, or contained the local router ID (see above) if the cluster ID is not configured. |
| Routes Advertised | The number of routes the device has advertised to this neighbor:<br>• To be Sent - The number of routes the device has queued to send to this neighbor.<br>• To be Withdrawn - The number of NLRIs for withdrawing routes the device has queued up to send to this neighbor in UPDATE messages. |
| NLRIs Sent in Update Message | The number of NLRIs for new routes the device has sent to this neighbor in UPDATE messages:<br>• Withdraws - The number of routes the device has sent to the neighbor to withdraw.<br>• Replacements - The number of routes the device has sent to the neighbor to replace routes the neighbor already has. |
| Peer Out of Memory Count for | Statistics for the times the device has run out of BGP4+ memory for the neighbor during the current BGP4+ session:<br>• Receiving Update Messages - The number of times UPDATE messages were discarded because there was no memory for attribute entries.<br>• Accepting Routes(NLRI) - The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count.<br>• Attributes - The number of times there was no memory for BGP4+ attribute entries.<br>• Outbound Routes (RIB-out) - The number of times there was no memory to place a "best" route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.<br>• Outbound Routes Holder - For debugging purposes only. |

# Examples

The following example displays routes summary information for neighbor 2001:db8::110.

```
device> show ipv6 bgp neighbor 2001:db8::110 routes-summary

1   IP Address: 2001:db8::110
Routes Accepted/Installed:0,  Filtered/Kept:0,  Filtered:0
   Routes Selected as BEST Routes:0
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0
NLRIs Received in Update Message:0,  Withdraws:0 (0),  Replacements:0
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
     Invalid Confed aspath:0, maxas-limit aspath:0
      Duplicated Originator_ID:0, Cluster_ID:0
Routes Advertised:2,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:2,  Withdraws:0,  Replacements:0
Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
   Attributes:0, Outbound Routes(RIB-out):0 Outbound Routes Holder:0
```

# show ipv6 bgp peer-group

Displays peer-group information.

## Syntax

**show ipv6 bgp peer-group** *peer-group-name*

## Parameters

*peer-group-name*
Specifies a peer group name.

## Modes

User EXEC mode

## Usage Guidelines

Only the parameters that have values different from their defaults are listed.

## Examples

The following example shows sample output from the **show ipv6 bgp peer-group** command.

```
device> show ipv6 bgp peer-group peer_group1

1   BGP peer-group is peer_group1
      Address family : IPV4 Unicast
        no activate
      Address family : IPV4 Multicast
        no activate
      Address family : IPV6 Unicast
        activate
      Address family : IPV6 Multicast
        no activate
      Address family : VPNV4 Unicast
        no activate
      Address family : L2VPN VPLS
        no activate
    Members:
       IP Address: 2000:400:400:400::3, AS: 65020
```

# show ipv6 bgp routes

Displays statistics for the routes in the device's BGP4+ route table.

## Syntax

**show ipv6 bgp routes** [ *num* | *ipv6-address/prefix* | **age** *num* | **as-path-access-list** *name* | **best** | **cidr-only** | **community-access-list** *name* | **community-reg-expression** *expression* | **detail** | **local** | **neighbor** *ipv6-addr* | **nexthop** *ipv6-addr* | **no-best** | **not-installed-best** | **prefix-list** *string* | **regular-expression** *name* | **route-map** *name* | **summary** | **unreachable** ]

## Parameters

*num*
Table entry at which the display starts. For example, if you want to list entries beginning with table entry 100, specify 100.

*ipv6-address/prefix*
Specifies an IPv6 address and prefix.

**age** *num*
Displays BGP4+ route information that is filtered by age.

**as-path-access-list** *name*
Displays BGP4+ route information that is filtered by autonomous system (AS)-path access control list (ACL).

**best**
Displays BGP4+ route information that the device selected as best routes.

**cidr-only**
Displays BGP4+ routes whose network masks do not match their class network length.

**community-access-list** *name*
Displays BGP4+ route information for an AS-path community access list.

**community-reg-expression** *expression*
Displays BGP4+ route information for an ordered community list regular expression.

**detail**
Displays BGP4+ detailed route information.

**local**
Displays BGP4+ route information about selected local routes.

**neighbor** *ipv6-addr*
Displays BGP4+ route information about selected BGP neighbors.

**nexthop** *ipv6-addr*
Displays BGP4+ route information about routes that are received from the specified next hop.

**no-best**
Displays BGP4+ route information that the device selected as not best routes.

**not-installed-best**
Displays BGP4+ route information about best routes that are not installed.

**prefix-list** *string*
Displays BGP4+ route information that is filtered by a prefix list.

**regular-expression** *name*
Displays BGP4+ route information about routes that are associated with the specified regular expression.

**route-map** *name*
Displays BGP4+ route information about routes that use the specified route map.

**summary**
Displays BGP4+ summary route information.

**unreachable**
Displays BGP4+ route information about routes whose destinations are unreachable through any of the BGP4+ paths in the BGP route table.

# Modes

User EXEC mode

# Command Output

The **show ipv6 bgp routes detail** command displays the following information:

| Output field | Description |
|---|---|
| Number of BGP4+ Routes | The number of routes displayed by the command. |
| Status codes | A list of the characters the display uses to indicate the route's status. The status code appears in the Status column of the display. The status codes are described in the command's output. |
| Prefix | The route's prefix. |
| Next Hop | For normal IPv6 routes, next hop is the next hop IPv6 router to reach the destination. For the 6PE routes, next hop is the IPv4-mapped IPv6 address of the peer 6PE router. |
| Metric | The value of the route's MED attribute. If the route does not have a metric, this field is blank. |
| LocPrf | The degree of preference for the advertised route relative to other routes in the local AS. When the BGP4+ algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 - 4294967295. |
| Weight | The value that this device associates with routes from a specific neighbor. For example, if the receives routes to the same destination from two BGP4+ neighbors, the prefers the route from the neighbor with the larger weight. |
| Status | The route's status, which can be one or more of the following:<br>• A - AGGREGATE. The route is an aggregate route for multiple networks.<br>• B - BEST. BGP4+ has determined that this is the optimal route to the destination.<br>• b - NOT-INSTALLED-BEST - BGP4+ has determined that this is the optimal route to the destination but did not install it in the IPv6 route table because the device received better routes from other sources (such as OSPFv3, RIPng, or static IPv6 routes).<br>• C - CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation.<br>• D - DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. |

| Output field | Description |
|---|---|
| | • E - EBGP. The route was learned through a in another AS.<br>• H - HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now.<br>• I - IBGP. The route was learned through a in the same AS.<br>• L - LOCAL. The route originated on this.<br>• M - MULTIPATH. BGP4+ load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B".<br>**NOTE**<br>If the "m" is shown in lowercase, the software was not able to install the route in the IPv6 route table.<br>• S - SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors. |
| AS-PATH | The AS-path information for the route. |

# Examples

The following example shows sample output from the **show ipv6 bgp routes** command.

```
device> show ipv6 bgp routes

Total number of BGP Routes: 4
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
       Prefix             Next Hop    MED     LocPrf    Weight Status
1      2001:db8:10:10::/64 ::         1                         100      32768  BL
         AS_PATH:
2      2001:db8:113:113::/64  ::
                                      1                         100      32768  BL
         AS_PATH:
3      2001::db8:400::/64  ::
                                      0        100      32768  BL
         AS_PATH:
4      2001:db8:400:400::/64  2001:db8:400:400::2
                                      0         400      0      I
         AS_PATH: 65005 65010
```

The following example shows sample output from the **show ipv6 bgp routes** command when the **detail** keyword is used.

```
device> show ipv6 bgp route detail

Total number of BGP Routes: 4
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH m:NOT-INSTALLED-MULTIPATH
       S:SUPPRESSED F:FILTERED s:STALE
1 Prefix: 2001:db8:10:10::/64,  Status: BL,  Age: 8h31m39s
       NEXT_HOP: ::, Learned from Peer: Local Router
       LOCAL_PREF: 100,  MED: 0,  ORIGIN: incomplete,  Weight: 32768
       AS_PATH:
       Adj_RIB_out count: 3,  Admin distance 1
2 Prefix: 2001:db8:113:113::/64,  Status: BL,  Age: 6h58m35s
       NEXT_HOP: ::, Learned from Peer: Local Router
       LOCAL_PREF: 100,  MED: 0,  ORIGIN: igp,  Weight: 32768
       AS_PATH:
       Adj_RIB_out count: 3,  Admin distance 1
3 Prefix: 2001:db8:202:202::/64,  Status: BI,  Age: 5h42m36s
       NEXT_HOP: 2001:db8:400:400::2, Metric: 0,  Learned from Peer: 2001:db8:400:400::2 (65020)
       LOCAL_PREF: 400,  MED: 0,  ORIGIN: incomplete,  Weight: 0
       AS_PATH: 65005 65010
       Adj_RIB_out count: 1,  Admin distance 200
4 Prefix: 2001:db8:400:400::/64,  Status: BL,  Age: 5h43m14s
       NEXT_HOP: ::, Learned from Peer: Local Router
       LOCAL_PREF: 100,  MED: 0,  ORIGIN: igp,  Weight: 32768
       AS_PATH:
       Adj_RIB_out count: 3,  Admin distance 1
```

# show ipv6 bgp routes community

Displays BGP4+ route information that is filtered by community and other options.

## Syntax

**show ipv6 bgp routes community** { *num* | *aa:nn* | **internet** | **local-as** | **no-advertise** | **no-export** }

## Parameters

**community**
Displays routes filtered by a variety of communities.

*num*
Specifies a community number n the range from 1 to 4294967200.

*aa:nn*
Specifies an autonomous system-community number.

**internet**
Displays routes for the Internet community.

**local-as**
Displays routes for a local sub-AS within the confederation.

**no-advertise**
Displays routes with this community that cannot be advertised to any other BGP4 devices at all.

**no-export**
Displays routes for the community of sub-ASs within a confederation.

## Modes

User EXEC mode

# show ipv6 bgp summary

Displays summarized information about the status of all BGP4+ connections.

## Syntax

**show ipv6 bgp summary**

## Modes

User EXEC mode

## Command Output

The **show ipv6 bgp summary** command displays the following information.

| Output field | Description |
|---|---|
| Router ID | The device's router ID. |
| Local AS Number | The BGP4+ AS number in which the device resides. |
| Confederation Identifier | The autonomous system number of the confederation in which the device resides. |
| Confederation Peers | The numbers of the local autonomous systems contained in the confederation. This list matches the confederation peer list you configure on the device. |
| Maximum Number of Paths Supported for Load Sharing | The maximum number of route paths across which the device can balance traffic to the same destination. The feature is enabled by default but the default number of paths is 1. You can increase the number from 2 - 8 paths. |
| Number of Neighbors Configured | The number of BGP4+ neighbors configured on this device. |
| Number of Routes Installed | The number of BGP4+ routes in the device's BGP4+ route table. |
| Number of Routes Advertising to All Neighbors | The total of the RtSent and RtToSend columns for all neighbors. |
| Number of Attribute Entries Installed | The number of BGP4+ route-attribute entries in the route-attributes table. |
| Neighbor Address | The IPv6 addresses of this BGP4+ neighbors. |
| AS# | The autonomous system number. |
| State | The state of this neighbor session with each neighbor. The states are from this perspective of the session, not the neighbor's perspective. The state values can be one of the following for each:<br>• IDLE - The BGP4+ process is waiting to be started. Usually, enabling BGP4+ or establishing a neighbor session starts the BGP4+ process.<br>– A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• ADMND - The neighbor has been administratively shut down.<br>– A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.<br>• CONNECT - BGP4+ is waiting for the connection process for the TCP neighbor session to be completed.<br>• ACTIVE - BGP4+ is waiting for a TCP connection from the neighbor. |

| Output field | Description |
|---|---|
| | **NOTE**<br>If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.<br>• OPEN SENT - BGP4+ is waiting for an Open message from the neighbor.<br>• OPEN CONFIRM - BGP4+ has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.<br>• ESTABLISHED - BGP4+ is ready to exchange UPDATE packets with the neighbor.<br>– If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed.<br>**NOTE**<br>If you display information for the neighbor using the **show ipv6 bgp neighbor***<ipv6-address>* command, the TCP receiver queue value will be greater than 0.<br>**Operational States:**<br>Additional information regarding the operational states of BGP described above may be added as described in the following:<br>• **(+)** - is displayed if there is more BGP data in the TCP receiver queue. **Note** : If you display information for the neighbor using the **show ip bgp neighbor***ip-addr* command, the TCP receiver queue value will be greater than 0.<br>• **(>)** - indicates that there is more BGP data in the outgoing queue.<br>• **(-)** - indicates that the session has gone down and the software is clearing or removing routes.<br>• **(*)** - indicates that the inbound or outbound policy is being updated for the peer.<br>• (**c)** - indicates that the table entry is clearing.<br>• (**p)** - indicates that the neighbor ribout group membership change is pending or in progress<br>• **(s)** - indicates that the peer has negotiated restart, and the session is in a stale state.<br>• **(r)** - indicates that the peer is restarting the BGP4 connection, through restart.<br>• **(^)** - on the standby MP indicates that the peer is in the ESTABLISHED state and has received restart capability (in the primary MP).<br>• **(<)** - indicates that the device is waiting to receive the "End of RIB" message the peer. |
| Time | The time that has passed since the state last changed. |
| Accepted | The number of routes received from the neighbor that this installed in the BGP4+ route table. Usually, this number is lower than the RoutesRcvd number. The difference indicates that this filtered out some of the routes received in the UPDATE messages. |
| Filtered | The routes or prefixes that have been filtered out.<br>• If soft reconfiguration is enabled, this field shows how many routes were filtered out (not placed in the BGP4+ route table) but retained in memory.<br>• If soft reconfiguration is not enabled, this field shows the number of BGP4+ routes that have been filtered out. |
| Sent | The number of BGP4+ routes that the has sent to the neighbor. |
| ToSend | The number of routes the has queued to send to this neighbor. |

# Examples

The following example displays sample output from the **show ipv6 bgp summary** command.

```
device> show ipv6 bgp summary

  device> show ipv6 bgp summary
BGP4 Summary
 Router ID: 113.1.1.1   Local AS Number: 65020
 Confederation Identifier: not configured
 Confederation Peers:
 Maximum Number of IP ECMP Paths Supported for Load Sharing: 1
 Number of Neighbors Configured: 2, UP: 1
 Number of Routes Installed: 5, Uses 430 bytes
 Number of Routes Advertising to All Neighbors: 7 (7 entries), Uses 336 bytes
 Number of Attribute Entries Installed: 4, Uses 360 bytes
 Neighbor Address     AS#   State  Time    Rt:Accepted Filtered Sent ToSend
 2001:db8:113:113::2  65001 CONN   1d14h32m       0      0     0      4
 2001:db8:400:400::2  65020 ESTAB  3h59m24s       2      0     3      0
```

# show ipv6 cache

Displays IPv6 cache information.

## Syntax

**show ipv6 cache** [ **vrf** *vrf-nam* ] [ *index* | *ipv6-address* | *ipv6-prefix/prefix-length* | **resource** | **ethernet** *stack/slot/port* | **tunnel** *tunnel-id* | **ve** *ve-num*]

## Parameters

**vrf** *vrf-name*

Displays the IPv6 cache information for the specified Virtual Routing/Forwarding (VRF) instance.

*index*

Restricts the display to the entry for the specified index number and subsequent entries.

*ipv6-address*

Restricts the display to the entries for the specified IPv6 address. Specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

*ipv6-prefix/prefix-length*

Restricts the display to the entries for the specified IPv6 prefix. Specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. Specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

**resource**

Displays the number of entries in the cache.

**ethernet** *stack/slot/port*

Restricts the display to the entries for the specified Ethernet interface.

**tunnel** *tunnel-id*

restricts the display to the entries for the specified tunnel interface.

**ve** *ve-num*

restricts the display to the entries for the specified VE interface.

## Modes

User EXEC mode

## Command Output

The **show ipv6 cache** command displays the following information:

| Output field | Description |
|---|---|
| Total number of cache entries | The number of entries in the cache table. |
| IPv6 Address | The host IPv6 address. |
| Next Hop | The next hop, which can be one of the following:<br>• Direct - The next hop is directly connected to the router. |

| Output field | Description |
|---|---|
| | • Local - The next hop is originated on this router.<br>• IPv6 address - The IPv6 address of the next hop. |
| Port | The port on which the entry was learned. |

# Examples

The following example displays the IPv6 cache information.

```
device# show ipv6 cache
Total number of cache entries: 10
   IPv6 Address                    Next Hop         Port
1  2001:DB8::2                     LOCAL            tunnel 2
2  2001:DB8::106                   LOCAL            ethe 1/3/2
3  2001:DB8::110                   DIRECT           ethe 1/3/2
4  2001:DB8:46a::1                 LOCAL            ethe 1/3/2
5  2001:DB8::2e0:52ff:fe99:9737    LOCAL            ethe 1/3/2
6  2001:DB8::ffff:ffff:feff:ffff   LOCAL            loopback 2
7  2001:DB8::c0a8:46a              LOCAL            tunnel 2
8  2001:DB8::c0a8:46a              LOCAL            tunnel 6
9  2001:DB8::1                     LOCAL            loopback 2
10 2001:DB8::2e0:52ff:fe99:9700    LOCAL            ethe 1/3/1
```

# show ipv6 dhcp-relay

Displays the DHCPv6 relay agent information configured on the device.

## Syntax

**show ipv6 dhcp-relay**

## Modes

Global configuration mode

## Usage Guidelines

## Command Output

The **show ipv6 dhcp-relay** command displays the following information:

| Output field | Description |
|---|---|
| Current DHCPv6 relay agent state | Displays whether the current relay agent state is enabled or disabled. |
| DHCPv6 enabled interface(s) | Displays the DHCPv6 enabled interfaces. |
| DHCPv6 Relay Agent Statistics | Displays statistics such as the total number of DHCPv6 packets received and transmitted. |
| Received DHCPv6 Packets | The number of release, relay forward and relay reply packets received. |

## Examples

The following example displays the IPv6 DHCP relay statistics.

```
device(config)# show ipv6 dhcp-relay
Current DHCPv6 relay agent state: Enabled
DHCPv6 enabled interface(s): e 1/1/3
DHCPv6 Relay Agent Statistics:
Total DHCPv6 Packets, Received:0, Transmitted:0
Received DHCPv6 Packets: RELEASE:0,RELAY_FORWARD:0,RELAY_REPLY:0
OtherServertoClient:0,OtherClinettoServer:0
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# show ipv6 dhcp-relay delegated-prefixes

Displays information about the IPv6 delegated prefixes.

## Syntax

**show ipv6 dhcp-relay delegated-prefixes interface** *interface-id*

## Parameters

**interface** *interface-id*

Displays delegated prefixes for the specified outgoing interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show ipv6 dhcp-relay delegated-prefixes** command displays the following information.

| Output field | Description |
|---|---|
| IPv6 Prefix | The IPv6 prefix delegated to the client. |
| Client | The IPv6 address of the client. |
| Interface | The interface on which the DHCPv6 messages are relayed to the client. |
| ExpireTime | The remaining lifetime of the delegated prefix. |

## Examples

The following example displays information about the delegated prefixes.

```
device# show ipv6 dhcp-relay delegated-prefixes interface ethernet 1/1/45

Prefix                      Client              Interface  ExpireTime
fc00:2000:6:7:1::/96        fe80::210:94ff:fe00:e 1/1/45   29d23h53m0s
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# show ipv6 dhcp-relay destinations

Displays the IPv6 DHCP relay destinations.

## Syntax

**show ipv6 dhcp-relay destinations**

## Modes

Global configuration mode

## Command Output

The **show ipv6 dhcp-relay destinations** command displays the following information:

| Output field | Description |
|---|---|
| DHCPv6 Relay Destinations | The DHCPv6 relay agent configured destination information. |

## Examples

The following example displays the IPv6 DHCP relay destinations.

```
device# show ipv6 dhcp-relay destinations
DHCPv6 Relay Destinations:
Interface e 1/2/3:
Destination OutgoingInterface
2001::2 NA
fe80::224:38ff:febb:e3c0 e 1/2/5
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# show ipv6 dhcp-relay interface

Displays the IPv6 DHCP relay information for a specific interface.

## Syntax

**show ipv6 dhcp-relay interface** *stack/slot/port*

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show ipv6 dhcp-relay interface** command displays the following information:

| Output field | Description |
|---|---|
| DHCPv6 Relay Information for *interface interface-type port-num* | The DHCPv6 relay information for the specific interface. |
| Destination | The configured destination IPv6 address. |
| OutgoingInterface | The interface on which the packet will be relayed if the destination relay address is a link local or multicast address. |
| Options | The current information about the DHCPv6 relay options for the interface. |
| Interface-Id | The interface ID option indicating whether the option is used. |
| Option-79 | Displays if option-79 is used or not. |

## Examples

The following example displays the DHCPv6 relay information for an interface.

```
device# show ipv6 dhcp-relay interface ethernet 1/1/1
DHCPv6 Relay Information for interface e 1/1/1:
Destinations:
 Destination                   OutgoingInterface
 2001::2                       NA
Options:
 Interface-Id: No     Remote-Id:No        Option-79:Yes
Prefix Delegation Information:
 Current:0 Maximum:100 AdminDistance:10
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |
| 8.0.40 | Support for link-layer-option (option 79) was introduced. |

# show ipv6 dhcp-relay options

Displays information about the relay options available to the prefixed delegates for a specific interface.

## Syntax

**show ipv6 dhcp-relay options**

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show ipv6 dhcp-relay options** command displays the following information:

| Output field | Description |
|---|---|
| Interface | The interface name. |
| Interface-Id | The interface ID option. Yes indicates the option is used; No indicates the option is not used. |
| Remote-Id | The remote ID option. Yes indicates the option is used; No indicates the option is not used. |
| Option-79 | The client link layer option. Yes indicates the option is used, No indicates the option is not used. |

## Examples

The following example displays relay options information.

```
device# show ipv6 dhcp-relay options
DHCPv6 Relay Options Information:
Interface       Interface-Id     Remote-Id     Option-79
e 1/1/1         No               No              No
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |
| 8.0.40 | Support for link-layer-option was added. |

# show ipv6 dhcp-relay prefix-delegation-information

Displays information about the IPv6 DHCP prefix delegation.

## Syntax

**show ipv6 dhcp-relay prefix-delegation-information**

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show ipv6 dhcp-relay prefix-delegation-information** command displays the following information:

| Output field | Description |
|---|---|
| Interface | The interface name. |
| Current | The number of delegated prefixes currently learned on the interface. |
| Maximum | The maximum number of delegated prefixes that can be learned on the interface. |
| AdminDistance | The current administrative distance used for prefixes learned on this interface when added to the IPv6 static route table. |

## Examples

The following example displays information about the IPv6 DHCP delegated prefixes.

```
device# show ipv6 dhcp prefix-delegation-information
DHCPv6 Relay Prefix Delegation Notification Information:
Interface      Current    Maximum     AdminDistance
ve 100          20          20000         10
ve 101         4000         20000         10
ve 102           0          20000         10
ve 103           0          20000         10
ve 104           0          20000         10
ve 105           0          20000         10
```

## History

| Release version | Command history |
|---|---|
| 08.0.10d | This command was introduced. |
| 08.0.30 | Support for this command was added in 08.0.30 and later releases. |

# show ipv6 dhcp6 snooping vlan

Displays the IPv6 DHCP snooping status on a VLAN.

## Syntax

**show ipv6 dhcp6 snooping vlan** *vlan-name*

## Parameters

*vlan-name*

The name of the VLAN.

## Modes

User EXEC mode

## Usage Guidelines

## Examples

The following example shows the status of DHCPv6 snooping enabled on VLAN 10.

```
device# show ipv6 dhcp6 snooping vlan 10
IP dhcpv6 snooping VLAN 10: Enabled
Trusted Ports: ethernet 1/1/1
Untrusted Ports: ethernet 1/1/2 ethernet 1/1/3
```

# show ipv6 dhcp6 snooping info

Displays the DHCPv6 snooping binding database.

## Syntax

**show ipv6 dhcp6 snooping info**

## Modes

User EXEC mode

## Usage Guidelines

Use this command to display information about the DHCPv6 snooping learnt entries.

## Examples

The following example shows the DHCPv6 snooping learnt entries.

```
device# show ipv6 dhcp6 snooping info
IPv6 Address LinkLayer-Addr  Age    Port        Virtual Port  vlan    VRF
 15::4       0000.0006.0003  259199  e 22/1/48    v15          15    default-vrf
 14::5       0000.0005.0004  259199  e 21/1/48    v14          14    default-vrf
 13::6       0000.0004.0005  259199  e 19/1/24    v13          13    default-vrf
 10::2       0000.0001.0001  259199  e 2/1/48     v10          10    default-vrf
 13::2       0000.0004.0001  259199  e 19/1/24    v13          13    default-vrf
 12::3       0000.0003.0002  259199  e 18/1/48    v12          12    default-vrf
 11::4       0000.0002.0003  259199  e 3/1/48     v11          11    default-vrf
 10::5       0000.0001.0004  259199  e 2/1/48     v10          10    default-vrf
 15::3       0000.0006.0002  259199  e 22/1/48    v15          15    default-vrf
 14::4       0000.0005.0003  259199  e 21/1/48    v14          14    default-vrf
 13::5       0000.0004.0004  259199  e 19/1/24    v13          13    default-vrf
 12::6       0000.0003.0005  259199  e 18/1/48    v12          12    default-vrf
 15::6       0000.0006.0005  259199  e 22/1/48    v15          15    default-vrf
 12::2       0000.0003.0001  259199  e 18/1/48    v12          12    default-vrf
 11::3       0000.0002.0002  259199  e 3/1/48     v11          11    default-vrf
 10::4       0000.0001.0003  259199  e 2/1/48     v10          10    default-vrf
 15::2       0000.0006.0001  259199  e 22/1/48    v15          15    default-vrf
 14::3       0000.0005.0002  259199  e 21/1/48    v14          14    default-vrf
 13::4       0000.0004.0003  259199  e 19/1/24    v13          13    default-vrf
 12::5       0000.0003.0004  259199  e 18/1/48    v12          12    default-vrf
 11::6       0000.0002.0005  259199  e 3/1/48     v11          11    default-vrf
 15::5       0000.0006.0004  259199  e 22/1/48    v15          15    default-vrf
 14::6       0000.0005.0005  259199  e 21/1/48    v14          14    default-vrf
 11::2       0000.0002.0001  259199  e 3/1/48     v11          11    default-vrf
 10::3       0000.0001.0002  259199  e 2/1/48     v10          10    default-vrf
 14::2       0000.0005.0001  259199  e 21/1/48    v14          14    default-vrf
 13::3       0000.0004.0002  259199  e 19/1/24    v13          13    default-vrf
 12::4       0000.0003.0003  259199  e 18/1/48    v12          12    default-vrf
 11::5       0000.0002.0004  259199  e 3/1/48     v11          11    default-vrf
 10::6       0000.0001.0005  259199  e 2/1/48     v10          10    default-
vrf

Total number of entries: 30
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This output of this command was modified. |

# show ipv6 fragment-header

Displays information about the current status of the IPv6 fragment header bit.

## Syntax

**show ipv6 fragment-header**

## Modes

User EXEC mode

## Examples

The following command shows that the virtual LAG specified by LAG ID 2 is not available in the system.

```
device> show ipv6 fragment-header

The fragment header bit ptb icmp is currently set
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | The command was introduced. |

# show ipv6 mld group

Displays the list of multicast listening discovery (MLD) groups.

## Syntax

**show ipv6 mld** [ **vrf** *vrf-name* ] **group** [ *ip-address* [ **detail** | **tracking** ] ]

## Parameters

**vrf** *vrf-name*
Displays information for the specified VRF instance.

*group-address*
Specifies the IPv6 address of the MLD group.

**detail**
Displays detailed information on the MLD group identified by the IPv6 address.

**tracking**
Displays information about who sends the reports.

## Modes

User EXEC mode

## Command Output

The **show ipv6 mld group** command displays the following information:

| Output Field | Description |
|---|---|
| IDX | Index for the MLD group. |
| Group Address | IPv6 address of the multicast group. |
| Port | The physical port to which the group belongs. |
| Intf | The routing interface to which the port belongs. |
| GrpCmpV | The version of the MLD group report message. |
| Mode | Indicates if the filter mode of the multicast group is in INCLUDE or EXCLUDE. |
| Timer | The number of seconds the interface can remain in its current mode. |
| Total number of groups | The total number of MLD groups. |

# Examples

The following example shows MLD statistics.

```
device> show ipv6 mld group
Total 2 groups
--------------------------------------------------------------------------------
Idx  Group Address                          Port    Intf   GrpCmpV Mode    Timer Srcs
----+---------------------------------------+------+------+-------+-------+-----+----
   1 ff05::4422                             e3/1/1  v170      Ver1 exclude   221   0
   2 ff3f::300                              e3/1/1  v170      Ver2 include     0   1
Total number of groups 2
```

# show ipv6 mld interface

Displays multicast listening discovery (MLD) parameters on an interface, including timers, the current querying router, and whether MLD is enabled.

## Syntax

**show ipv6 mld** [ **vrf** *vrf-name* ] **interface** [ **ethernet** *stack/slot/port* | **ve** *ve-num* [ **group** *A.B.C.D* ] | **tunnel** *tunnel-id* ]

## Parameters

**vrf** *vrf-name*
Displays information for the specified VRF instance.

**ethernet** *stack/slot/port*
Displays information for a specific Ethernet interface.

**ve** *num*
Displays nformation for a specific VE interface.

**group** *A.B.C.D*
Specifies displaying information for a specific group address.

**tunnel** *tunnel-id*
Displays information for a specific Tunnel interface.

## Modes

User EXEC mode

## Command Output

The **show ipv6 mld interface** command displays the following information:

| Output Field | Description |
|---|---|
| version | Version of the MLD being used. |
| query int | Query interval in seconds. |
| max resp time | Number of seconds multicast groups have to respond to queries. |
| group mem time | Number of seconds multicast groups can be members of this group before aging out. |
| (details) | The following is displayed for each interface:<br>• The port ID<br>• The default MLD version being used<br>• The multicast protocol used<br>• IPV6 address of the multicast interface<br>• If the interface has groups, the group source list, IPv6 multicast address, and the filter mode are displayed. |

# Examples

The following example shows MLD statistics for an interface.

```
device# show ipv6 mld interface
---------+------+---------+-----------------------------------------+--
Intf/Port|Groups| Version |   Querier      | Timer |V1Rtr| Tracking
                |Oper Cfg |  Qrr   GenQ    |             |
---------+------+----+----+-----------------------------------------+--
e1/1/1      0     2    -    Self     0         0     No     Disabled
v40         0     2    -                                    Disabled
e3/1/1      1     2    -    Self     0         0     No
e2/1/1      1     2    -    Self     0         0     No
e1/1/1      1     2    -    Self     0         0     No
v50         0     2    -                                    Disabled
e1/1/2      0     2    -    Self     0         0     No
v220        0     2    -                                    Disabled
e1/1/1      3     2    -    Self     0         12    No
```

The following example shows MLD statistics on an interface for a VRF named my_vrf.

```
device# show ipv6 mld vrf my_vrf interface
---------+------+---------+-----------------------------------------+---------+-----+---------
Intf/Port|Groups| Version |           Querier                       |  Timer  |V1Rtr|Tracking
         |      |Oper  Cfg| OQrr                      GenQ          |         |     |
---------+------+----+----+-----------------------------------------+----+----+-----+---------
v6          0    2    -                                                          Disabled
e1/1/1      2         -    fe80::20c:dbff:fee2:5000   11              0      No
v61         0    2    -                                                          Disabled
e1/1/2      2         -    Self                         0             122    No
```

The following example displays information for the interface VE 4041 group.

```
 device# show ipv6 mld interface ve 4041 group
Total 1 groups
-----------------------------------------------------------------
Idx  Group Address  Port   Intf      GrpCmpV  Mode     Timer Srcs
----+---------------------------------+---------+---------
 1    ff1e::6:1    e1/2/8   v4041     Ver1   exclude  178   0

Total number of groups 1
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to include the MLD **group** keyword. |

# show ipv6 mld settings

Displays multicast listening discovery (MLD) settings.

## Syntax

**show ipv6 mld** [ **vrf** *vrf-name* ] **settings**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 mld settings** command displays the following information:

| Output Field | Description |
|---|---|
| Query Interval | How often the router will query an interface for group membership. |
| Configured Interval | The interval that has been configured for the router. |
| Max Response Time | The length of time in seconds that the router will wait for an IGMP (V1 or V2) response from an interface before concluding that the group member on that interface is down and removing it from the group. |
| Group Membership Time | The length of time in seconds that a group will remain active on an interface in the absence of a group report. |
| Operating Version | The IGMP version operating on the router. |
| Configured Version | The IGMP version configured on the router. |
| Robustness Variable | Used to fine-tune for unexpected loss on the subnet. The value is used to calculate the group interval. |
| Last Member Query Interval | Indicates when a leave is received; a group-specific query is sent. The last member query count is the number of queries with a time interval of (LMQT) is sent. |
| Last Member Query Count | Specifies the number of group-specific queries when a leave is received. |

# Examples

The following example shows MLD settings for the VRF named my_vrf.

```
device# show ipv6 mld vrf my_vrf settings
MLD Global Configuration
  Query Interval            : 125s   Configured Interval    : 125s
  Max Response Time         : 10s
  Group Membership Time     : 260s
  Operating Version         : 2      Configured Version     : 0
  Robustness Variable       : 2
  Last Member Query Interval: 1s      Last Member Query Count: 2
  Older Host Present Timer  : 260s
```

# show ipv6 mld static

Displays static multicast listening discovery (MLD) groups.

## Syntax

**show ipv6 mld** [ **vrf** *vrf-name* ] **static**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 mld static** command displays the following information:

| Output Field | Description |
|---|---|
| Group Address | The address of the multicast group. |
| Interface Port List | The physical ports on which the multicast groups are received. |

## Examples

The following example shows MLD settings for the VRF named my_vrf.

```
device# show ipv6 mld vrf my_vrf static
Group Address                           Interface Port List
----------------------------------------+---------+---------
ff1e:1::1                               v3       ethe 1/2/10
ff1e:a::7f                              v3       ethe 1/2/10
```

# show ipv6 mld traffic

Displays information about multicast listening discovery (MLD) traffic.

## Syntax

**show ipv6 mld** [ **vrf** *vrf-name* ] **traffic**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 mld traffic** command displays the following information:

| Output Field | Description |
|---|---|
| QryV1 | Number of general MLDv1 queries received or sent by the virtual routing interface. |
| QryV2 | Number of general MLDv2 queries received or sent by the virtual routing interface. |
| G-Qry | Number of group-specific queries received or sent by the virtual routing interface. |
| GSQry | Number of source specific queries received or sent by the virtual routing interface. |
| MbrV1 | Number of MLDv1 membership reports received. |
| MbrV2 | Number of MLDv2 membership reports received. |
| Leave | Number of MLDv1 "leave" messages on the interface. (See 2_Ex for MLDv2.) |
| Is_IN | Number of source addresses that were included in the traffic. |
| Is_EX | Number of source addresses that were excluded in the traffic. |
| ToIN | Number of times the interface mode changed from exclude to include. |
| ToEX | Number of times the interface mode changed from include to exclude. |
| ALLOW | Number of times that additional source addresses were allowed or denied on the interface. |
| BLK | Number of times that sources were removed from an interface. |

# Examples

The following example shows MLD traffic.

```
device# show ipv6 mld traffic
Recv     QryV1 QryV2 G-Qry GSQry MbrV1 MbrV2 Leave IS_IN IS_EX ToIN ToEX ALLO  BLK
e1/3/1       0     0     0     0     0     0     0     0     0    0    0    0    0
e1/3/2       0     0     0     0     0     0     0     0     0    0    0    0    0
e1/6/18      0     0     0     0     0   176     0   110     0    0    0   66    0
e1/6/19      0     0     0     0     0   176     0   110     0    0    0   66    0
e1/6/20      0     0     0     0     0   176     0   110     0    0    0   66    0
e1/6/25      0     0     0     0     0   176     0   110     0    0    0   66    0
l1           0     0     0     0     0     0     0     0     0    0    0    0    0
Send     QryV1 QryV2 G-Qry GSQry
e1/3/1       0     0     0     0
e1/3/2       0     0     0     0
e1/6/18      0    10    10     0
e1/6/19      0    10    10     0
e1/6/20      0    10    10     0
e1/6/25      0    10    10     0
l1           0     0     0     0
```

# show ipv6 mroute

Displays information on IPv6 multicast routes. You can specify displaying information either from static or connected mroutes or from a particular mroute.

## Syntax

**show ipv6 mroute** [**vrf** *vrf-name* ] { *ipv6-address ipv6-prefix/prefix-length* | **static** | **connect** | **summary** }

## Parameters

**vrf** *vrf-name*
Specifies displaying mroutes for a particular VRF.

*ipv6-address ipv6-prefix/prefix-length*
Displays an IPv6 mroute for the specified destination.

**static**
Displays only static multicast routes.

**connect**
Displays only connected multicast routes.

**summary**
Displays summary information.

## Modes

Privileged EXEC mode

## Examples

The following example displays information for IPv6 multicast routes:

```
Device(config)# show ipv6 mroute
IPv6 Routing Table - 7 entries:
Type Codes - B:BGP C:Connected S:Static
Type IPv6 Prefix           Next Hop Router    Interface     Dis/Metric     Uptime
S    1:1::1:0/120          ::                 ve 90         1/1            2d16h
C    2090::/64             ::                 ve 90         0/0            6d21h
C    2100::/64             ::                 ve 100        0/0            1d21h
C    2110::/64             ::                 ve 110        0/0            1d21h
C    2120::/64             ::                 ve 120        0/0            1d21h
C    2130::/64             ::                 ve 130        0/0            6d21h
C    8811::1/128           ::                 loopback 1    0/0            6d21h
```

The following example displays information for static IPv6 multicast routes:

```
Device(config)# show ipv6 mroute static
Type Codes - B:BGP C:Connected S:Static
Type IPv6 Prefix           Next Hop Router    Interface     Dis/Metric     Uptime
S    1:1::1:0/120          ::                 ve 90         1/1            2d16h
```

The following example displays information for directly attached (connected) IPv6 multicast routes:

```
Device(config)#show ipv6 mroute connect
Type Codes - B:BGP C:Connected S:Static
Type IPv6 Prefix           Next Hop Router    Interface     Dis/Metric     Uptime
C    2090::/64             ::                 ve 90         0/0            6d21h
C    2100::/64             ::                 ve 100        0/0            1d21h
C    2110::/64             ::                 ve 110        0/0            1d21h
C    2120::/64             ::                 ve 120        0/0            1d21h
C    2130::/64             ::                 ve 130        0/0            6d21h
C    8811::1/128           ::                 loopback 1    0/0            6d21h
```

The following example displays information for IPv6 multicast route 2090::1:

```
Device(config)# show ipv6 mroute 2090::1
Type Codes - B:BGP C:Connected S:Static
Type IPv6 Prefix           Next Hop Router    Interface     Dis/Metric     Uptime
C    2090::/64             ::                 ve 90         0/0            6d21h
```

# History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# show ipv6 multicast

Displays IPv6 IGMP snooping information.

## Syntax

**show ipv6 multicast**

## Modes

User EXEC mode

## Usage Guidelines

You can use the **show ipv6 multicast** command to display information for VLANs.

## Examples

The following example shows IGMP snooping information.

```
device# show ipv6 multicast vlan 4050
Version=1, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=255,
                      Leave Wait=2, Robustness=2

VL4050: dft V1, glb cfg passive, , pimsm (glb cfg), 0 grp, 1 (*G) cache, rtr ports,
    router ports: e2/1/6(200) fe80::4050:3, e1/1/4(85) fe80::4050:5,
  My Query address: fe80::ce4e:24ff:fe6f:980 (link-local)
  e1/1/1 has 0 grp, non-QR (QR=fe80::4050:3, age=40), dft V1 trunk
  e1/1/4 has 0 grp, non-QR (passive), dft V1 trunk
  e1/1/41 has 0 grp, non-QR (passive), dft V1 trunk
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | The output of this command was modified to display the robustness variable and leave-wait timer. |

# show ipv6 multicast error

Displays information about possible multicast listening discovery (MLD) errors.

## Syntax

**show ipv6 multicast error**

## Modes

User EXEC mode

## Command Output

The **show ipv6 multicast error** command displays the following information:

| Output Field | Description |
|---|---|
| SW processed pkt | The number of IPv6 multicast packets processed by MLD snooping. |
| up-time | The MLD snooping up time. |

## Examples

The following example shows information about possible MLD errors.

```
device# show ipv6 multicast error
snoop SW processed pkt: 173, up-time 160 sec
```

# show ipv6 multicast group

Displays information about multicast listening discovery (MLD) groups.

## Syntax

**show ipv6 multicast group** [*group-address* [**detail**] [**tracking**]]

## Parameters

*group-address*
Specifies information for a particular group.

**detail**
Specifies the source list of a specific VLAN.

**tracking**
Specifies tracking information on interfaces that have tracking enabled.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 multicast group** command displays the following information:

| Output Field | Description |
|---|---|
| group | The address of the IPv6 group (destination IPv6 address). |
| p-port | The physical port on which the group membership was received. |
| ST | **Yes** indicates that the MLD group was configured as a static group; **No** means it was learned from reports. |
| QR | **Yes** means the port is a querier port; **No** means it is not. A port becomes a non-querier port when it receives a query from a source with a lower source IP address than the port. |
| life | The number of seconds the group can remain in EXCLUDE mode. An EXCLUDE mode changes to INCLUDE if it does not receive an IS_EX or TO_EX message during a specified period of time. The default is 140 seconds. There is no `life` displayed in INCLUDE mode. |
| mode | The current mode of the interface: INCLUDE or EXCLUDE. If the interface is in INCLUDE mode, it admits traffic only from the source list. If the interface is in EXCLUDE mode, it denies traffic from the source list and accepts the rest. |
| source | Identifies the source list that will be included or excluded on the interface.<br>An MLDv1 group is in EXCLUDE mode with a source of 0. The group excludes traffic from 0 (zero) source list, which actually means that all traffic sources are included. |

| Output Field | Description |
|---|---|
| group | If you requested a *detailed* report, the following information is displayed:<br>• The multicast group address<br>• The mode of the group<br>• Sources from which traffic will be admitted (INCLUDE) or denied (EXCLUDE) on the interface.<br>• The life of each source list.<br>If you requested a *tracking/fast leave* report, the clients from which reports were received are identified. |

# Examples

This example shows that an MLDv1 group is in EXCLUDE mode with a source of 0. The group excludes only traffic from the 0 (zero) source list, which means that all traffic sources are included.

```
Device#show ipv6 multicast group
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL1 : 263 grp, 263 grp-port, tracking_enabled
      group                                   p-port ST QR life mode source
1     ff0e::ef00:a0e3                         1/7    N  Y  120  EX   0
2     ff01::1:f123:f567                       1/9    N  Y       IN   1
```

This example displays detailed MLD group information for multicast group ff0e::ef00:a096:

```
Device#show ipv6 multicast group ff0e::ef00:a096 detail
Display group ff0e::ef00:a096 in all interfaces in details.
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL1 : 1 grp, 1 grp-port, tracking_enabled
      group                                   p-port ST QR life mode source
1     ff0e::ef00:a096                         1/7    N  Y  100  EX   0
    group: ff0e::ef00:a096, EX, permit 0 (source, life):
      life=100, deny 0:
```

This example displays the list of clients that belong to multicast group ff0e::ef00:a096 when tracking and fast leave are enabled:

```
Device#show ipv6 multicast group ff0e::ef00:a096 tracking
Display group ff0e::ef00:a096 in all interfaces with tracking enabled.
p-:physical, ST:static, QR:querier, EX:exclude, IN:include, Y:yes, N:no
VL1 : 1 grp, 1 grp-port, tracking_enabled
      group                                   p-port ST QR life mode source
1     ff0e::ef00:a096                         1/7    N  Y  80   EX   0
    receive reports from 1 clients: (age)
      (2001:DB8::1011:1213:1415 60)
```

# show ipv6 multicast mcache

Displays information in the IPv6 multicast forwarding mcache (multicast listening discovery [MLD]).

## Syntax

**show ipv6 multicast mcache**

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 multicast mcache** command displays the following information:

| Output Field | Description |
|---|---|
| (abcd:ef50 0:100): | The lowest 32 bits of source and group. It is displayed in XXXX:XXXX hex format. Here XXXX is a 16-bit hex number. |
| cnt | The number of packets processed in software. |
| OIF | Output interfaces. |
| age | The mcache age in seconds. The mcache is reset to 0 if traffic continues to arrive, otherwise it is aged out when it reaches the time defined by the **ipv6 multicast mcache-age** command. |
| uptime | The up time of this mcache in seconds. |
| vidx | The vidx is shared among mcaches using the same output interfaces. The vidx specifies the output port list, which shows the index. Valid range is from 4096 to 8191. |
| ref-cnt | The number of mcaches using this vidx. |

## Examples

This example shows information in the multicast forwarding mcache:

```
Device#show ipv6 multicast mcache
Example: (S G) cnt=: (S G) are the lowest 32 bits, cnt: SW proc. count
    OIF: 1/1/22 TR(1/1/32,1/1/33), TR is trunk, 1/1/32 primary, 1/33 output
vlan 1, has 2 cache
1     (abcd:ef50 0:100), cnt=121
      OIF: 1/1/11 1/1/9
      age=0s up-time=120s vidx=4130 (ref-cnt=1)
2     (abcd:ef50 0:101), cnt=0
      OIF: entire vlan
      age=0s up-time=0s vidx=8191 (ref-cnt=1)
vlan 70, has 0 cache
```

# show ipv6 multicast optimization

Displays multicast listening discovery (MLD) snooping hardware resource-sharing information.

## Syntax

**show ipv6 multicast optimization** [ *l2mc* ]

## Parameters

*l2mc*

Specifies the Layer 2 multicast (L2MC) group index.

## Modes

Privileged EXEC mode

VLAN configuration mode

## Usage Guidelines

The **show ipv6 multicast optimization** command is supported only on the ICX 7250, ICX 7450, and ICX 7750 devices.

Use this command to display the availability of L2MC group indexes in the hardware and how it is used and shared

The L2MC group index range varies depending on the platform. Values out of range are not displayed.

## Examples

The following example displays resource information showing that L2MC group index 4 is shared by two users and the ports included in the set are 1/1/6 and 1/1/1:

```
Device (config)# vlan 150
Device (config-vlan-150)# show ipv6 multicast optimization
Total L2MCs Allocated:    0; Available: 8192; Failed:    0
Index    L2MC         SetId         Users         Set
   1.    4             0x161fcbd8        2 {<1/1/6>,<1/1/1>,}
   2.    1             0x161d0930       10 {<1/1/6>,<1/1/4>,<1/1/3>,<1/1/2>,
                                           <1/1/1>,}
Sharability Coefficient:  76%
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# show ipv6 multicast pimsm-snooping

Displays information related to PIM sparse mode (SM) snooping on the mcache.

## Syntax

**show ipv6 multicast pimsm-snooping** [ **vlan** *vlan-id* ] [ **cache** *ipv6-address* ] [ **resources** ]

## Parameters

**cache** *ipv6-address*
Specifies the PIM SM Snooping cache.

**vlan** *vlan-id*
Specifies snooping for a VLAN.

**resources**
Specifies PIM SM snooping resources.

## Modes

Privileged exec mode

## Usage Guidelines

Use the **show ipv6 pimsm-snooping cache** command to display information related to the PIM SM snooping outgoing interface (OIF) in the mcache.

## Examples

The following example shows PIM SM information for the mcache:

```
Device#show ipv6 multicast pimsm-snooping
Example: Port: 7/3 (ref_count=1)
         ref_count: no of entries in pimsm snoop cache added this oif)

vlan 503, has 1 caches.
1    (* 2:3) has 1 pim join ports out of 1 OIF
     1/1/4 (ref_count=2),
```

# show ipv6 multicast resource

Displays information about the software resources used.

## Syntax

**show ipv6 multicast resource** [ **vlan** *vlan-num* ]

## Parameters

**vlan** *vlan-num*

Displays information for the specified VLAN only.

## Modes

User EXEC mode

## Command Output

The **show ipv6 multicast resource** command displays the following information:

| Output Field | Displays |
|---|---|
| alloc | The allocated number of units. |
| in-use | The number of units currently used. |
| avail | The number of available units. |
| get-fail | The number of resource failures<br><br>**NOTE**<br>It is important to pay close attention to this field. |
| limit | The upper limit of this expandable field. The multicast listening discovery (MLD) group limit is configured using the **system-max mld-snoop-group-addr** command. The snoop mcache entry limit is configured using the **system-max mld-snoop-mcache** command. |
| get-mem | The current memory allocation. This number should continue to increase. |
| size | The size of a unit (in bytes). |
| init | The initial allocated amount of memory.<br><br>**NOTE**<br>This number can be increased. (More memory can be allocated if necessary.) |
| Available vidx | The output interface (OIF) port mask used by mcache. The entire device has a maximum of 4096 vidx. Different mcaches with the same OIF share the same vidx. If vidx is not available, the stream cannot be hardware-switched. |

# Examples

This example shows information about the software resources:

```
device> show ipv6 multicast resource
                     alloc in-use  avail get-fail    limit  get-mem  size init
mld group              512      9    503        0    32000      272    28  256
mld phy port          1024     16   1008        0   200000      279    21 1024
snoop group hash       512      9    503        0    59392      272    20  256
.... Entries deleted
total pool memory 194432 bytes
has total 1 forwarding hash
Available vidx: 4061
```

# show ipv6 multicast traffic

Displays status information for multicast listening discovery (MLD) snooping traffic.

## Syntax

**show ipv6 multicast traffic**

## Modes

User EXEC mode

## Command Output

The **show ipv6 multicast traffic** command displays the following information:

| Output Field | Description |
|---|---|
| Q | Query |
| Qry | General Query |
| QryV1 | Number of general MLDv1 queries received or sent. |
| QryV2 | Number of general MLDv2 snooping queries received or sent. |
| G-Qry | Number of group specific queries received or sent. |
| GSQry | Number of group source specific queries received or sent. |
| MBR | The membership report. |
| MbrV1 | The MLDv1 membership report. |
| MbrV2 | The MLDv2 membership report. |
| IsIN | Number of source addresses that were included in the traffic. |
| IsEX | Number of source addresses that were excluded in the traffic. |
| ToIN | Number of times the interface mode changed from EXCLUDE to INCLUDE. |
| ToEX | Number of times the interface mode changed from INCLUDE to EXCLUDE. |
| ALLO | Number of times additional source addresses were allowed on the interface. |
| BLK | Number of times sources were removed from an interface. |
| Pkt-Err | Number of packets having errors such as checksum errors. |

# Examples

The following example shows information for MLD snooping traffic.

```
device> show ipv6 multicast traffic
MLD snooping: Total Recv: 32208, Xmit: 166
Q: query, Qry: general Q,  G-Qry: group Q,  GSQry: group-source Q, Mbr: member
Recv       QryV1      QryV2      G-Qry      GSQry      MbrV1      MbrV2      Leave
VL1            0          0          0          0      31744        208        256
VL70           0          0          0          0          0          0          0
Recv        IsIN       IsEX       ToIN       ToEX      ALLOW      BLOCK    Pkt-Err
VL1         1473      31784          0          1          1          7          0
VL70           0          0          0          0          0          0          0
Send       QryV1      QryV2      G-Qry      GSQry      MbrV1      MbrV2
VL1            0          0        166          0          0          0
VL70           0          0          0          0          0          0
```

# show ipv6 multicast vlan

Displays multicast listening discovery (MLD) snooping information for all VLANs or for a specific VLAN.

## Syntax

**show ipv6 multicast vlan** *vlan-id*

## Parameters

*vlan-id*

Specifies the VLAN for which you want information. If you do not specify a *vlan-id*, information for all VLANs is displayed.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 multicast vlan** command displays the following information:

| Output Field | Description |
|---|---|
| version | The MLD version number. |
| query-t | How often a querier sends a general query on the interface. |
| group-aging-t | Number of seconds membership groups can be members of this group before aging out. |
| rtr-port | The router ports which are the ports receiving queries. The display router ports: 1/36(120) 2001:DB8::2e0:52ff:fe00:9900 means port 1/36 has a querier with 2001:DB8::2e0:52ff:fe00:9900 as the link-local address, and the remaining life is 120 seconds. |
| max-resp-t | The maximum number of seconds a client can wait before it replies to the query. |
| non-QR | Indicates that the port is a non-querier. |
| QR | Indicates that the port is a querier. |
| Unregistered IPv6 Multicast Packets Flooding | Indicates whether flooding is enabled. |

## Examples

The following example shows MLD snooping information for VLAN 70:

```
Device#show ipv6 multicast vlan 70
version=1, query-t=60, group-aging-t=140, max-resp-t=3, other-qr-present-t=123
VL70: cfg V2, vlan cfg passive, 2 grp, 0 (SG) cache, rtr ports,
    router ports: 1/36(120) 2001:DB8::2e0:52ff:fe00:9900,
  My Query address: fe80::ce4e:24ff:fe6f:980 (link-local)
  1/26 has 2 grp, non-QR (passive), cfg V1
  1/26 has 2 grp, non-QR (passive), cfg V1
    group: ff10:1234::5679, life = 100
    group: ff10:1234::5678, life = 100
  1/35 has 0 grp, non-QR (QR=2001:DB8::2e0:52ff:fe00:9900, age=20), dft V2 trunk
```

The following example shows MLD snooping information when flooding of unregistered IPv6 multicast frames is disabled:

```
Device#show ipv6 multicast vlan
Summary of all vlans. use "sh ipv6 multicast vlan vlan-id" for details
Version=1, Intervals: Query=125, Group Age=260, Max Resp=10, Other Qr=255
Leave Wait=2, Robustness=2
Unregistered IPv6 Multicast Packets Flooding: Disabled.

VL500: dft V1, vlan cfg active, 0 grp, 0 (*G) cache, no rtr port,
VL600 no snoop: no global or local config
My Query address: fe80::ce4e:24ff:fe6f:980 (link-local)
```

# History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was modified to display flooding information. |
| 8.0.50 | The output of this command was modified to display the My Query address field. |

# show ipv6 neighbor

Displays the IPv6 neighbor table, which contains an entry for each IPv6 neighbor with which the router exchanges IPv6 packets.

## Syntax

**show ipv6 neighbor** [ **vrf** *vrf-nam* ] [ *index* | *ipv6-address* | *ipv6-prefix/prefix-length* | **ethernet** *stack**/**slot**/**port* | **ve** *ve-num*]

## Parameters

**vrf** *vrf-name*

Displays the IPv6 neighbor information for the specified Virtual Routing/Forwarding (VRF) instance.

*ipv6-address*

Restricts the display to the entries for the specified IPv6 address. Specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

*ipv6-prefix/prefix-length*

Restricts the display to the entries for the specified IPv6 prefix. Specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. Specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

**ethernet** *stack**/**slot**/**port*

Restricts the display to the entries for the specified Ethernet interface.

**ve** *ve-num*

restricts the display to the entries for the specified VE interface.

## Modes

User EXEC mode

## Command Output

The **show ipv6 neighbor** command displays the following information:

| Output field | Description |
|---|---|
| Total number of neighbor entries | The total number of entries in the IPv6 neighbor table. |
| IPv6 Address | The 128-bit IPv6 address of the neighbor. |
| Link-Layer Address | The 48-bit interface ID of the neighbor. |
| State | The current state of the neighbor. Possible states are as follows:<br>• INCOMPLETE - Address resolution of the entry is being performed.<br>• *REACH - The static forward path to the neighbor is functioning properly.<br>• REACH - The forward path to the neighbor is functioning properly.<br>• STALE - This entry has remained unused for the maximum interval of two hours. While stale, no action takes place until a packet is sent. |

| Output field | Description |
|---|---|
| | • DELAY - This entry has remained unused for the maximum interval, and a packet was sent before another interval elapsed.<br>• PROBE - Neighbor solicitation are transmitted until a reachability confirmation is received. |
| Age | The number of seconds the entry has remained unused. If this value remains unused for the number of seconds specified by the ipv6 nd reachable-time command (the default is 30 seconds), the entry is removed from the table. |
| Port | The physical port on which the entry was learned. |
| vlan | The VLAN on which the entry was learned. |
| IsR | Determines if the neighbor is a router or host:<br>• 0 - Indicates that the neighbor is a host.<br>• 1 - Indicates that the neighbor is a router. |

**NOTE**
Oldest Stale entry will be deleted before the default time interval of two hours if the total number of entries in the neighbor table is equal to the maximum number of neighbor entries when a new entry is trying to be added.

# Examples

The following example displays the IPv6 neighbor table.

```
device> show ipv6 neighbor
Total number of Neighbor entries: 3
IPv6 Address              LinkLayer-Addr   State     Age    Port       vlan    IsR
2001:DB8::55              0000.0002.0002   *REACH    0      e 1/3/11   -       0
2000:4::110               0000.0091.bb37    REACH    20     e 1/3/1    5       1
fe80::2e0:52ff:fe91:bb37  0000.0091.bb37    DELAY    1      e 1/3/2    4       1
fe80::2e0:52ff:fe91:bb40  0000.0091.bb40    STALE    5930   e 1/3/3    5       1
```

# show ipv6 neighbor inspection

Displays the status of the neighbor discovery (ND) inspection configuration, details of the VLANs on which ND inspection is enabled, ND static entries, and ND inspection statistics.

## Syntax

**show ipv6 neighbor** [ **vrf** *vrf-name* ] **inspection** [**static-entry** | **statistics** | **vlan** *vlan-number* ]

## Parameters

**static-entry**

Specifies the manually configured static ND inspection entries that are used to validate the packets received on untrusted ports.

**statistics**

Specifies the total number of neighbor discovery messages received and the number of packets discarded after ND inspection.

**vlan**

Specifies the VLANs on which ND inspection is enabled.

*vlan-number*

Specifies the ID of the configured VLAN.

**vrf**

Specifies the VRF instance.

*vrf-name*

Specifies the ID of the VRF instance.

inspection

Specifies that the neighbor discovery messages are verified against the static ND inspection entries or dynamically learned DHCPv6 snoop entries.

## Modes

Privileged EXEC mode

Global configuration mode

VRF configuration mode

## Command Output

The **show ipv6 neighbor inspection** command displays the following information.

| Output field | Description |
|---|---|
| VLAN | The list of VLANs on which ND inspection is enabled. |
| IPv6 Address | The IPv6 addresses of the hosts that are added as static ND inspection entries. |
| LinkLayer-Addr | The MAC addresses of the hosts that are added as static ND inspection entries. |

| Output field | Description |
|---|---|
| Total number of ND Solicit received | The total number of neighbor solicitation messages received. |
| Total number of ND Advert received | The total number of neighbor advertisement messages received. |
| Total number of Router Solicit received | The total number of router solicitation messages received. |
| Total number of ND dropped | The total number of neighbor discovery messages that are discarded because of the IP-to-MAC address binding discrepancy. |
| IPv6 Neighbor inspection VLAN *vlan-number* | The status of ND inspection on a VLAN. |
| Untrusted Ports | The interfaces or member ports on which trust mode is not enabled. |
| Trusted Ports | The interfaces or member ports on which trust mode is enabled. |

# Examples

The following example shows the output of the **show ipv6 neighbor inspection** command.

```
device(config)# show ipv6 neighbor inspection
IPv6 Neighbor inspection enabled on 2 VLAN(s):
        VLAN: 2
        VLAN: 3
```

The following example shows the output of the ND inspection configuration details for a VRF.

```
device(config-vrf-3)# show ipv6 neighbor vrf 3 inspection
IPv6 Neighbor inspection enabled on 2 VLAN(s):
        VLAN: 2
        VLAN: 3
```

The following example shows the output of the **show ipv6 neighbor inspection static-entry** command.

```
device(config)# show ipv6 neighbor inspection static-entry
 Total number of ND Inspect entries: 3
 IPv6 Address                           LinkLayer-Addr
 2001::1                                0000.0000.1234
 2001::3                                0000.1234.4567
 2001::2                                0000.0000.4567
```

The following example shows the ND static entries of a VRF.

```
device(config-vrf-3)# show ipv6 neighbor vrf 3 inspection static-entry
 Total number of ND Inspect entries: 1
 IPv6 Address                           LinkLayer-Addr
 2001:201:1:1::34                        cc4e.246d.2038
```

The following example shows the output of the **show ipv6 neighbor inspection statistics** command.

```
device(config)# show ipv6 neighbor inspection statistics
Total number of ND Solicit received      11
Total number of ND Advert received      29
Total number of Router Solicit received  20
Total number of ND dropped               6
```

The following example shows the ND inspection statistics of a VRF.

```
device(config-vrf-3)# show ipv6 neighbor vrf 3 inspection statistics
Total number of ND Solicit received      11
Total number of ND Advert received      29
Total number of Router Solicit received  20
Total number of ND dropped               6
```

The following example shows the output of the **show ipv6 neighbor inspection vlan** *vlan-number* command.

```
device (config)# show ipv6 neighbor inspection vlan 2
IPv6 Neighbor inspection VLAN 2: Enabled
  Untrusted Ports : ethe 1/1/1 to 1/1/2
  Trusted Ports : ethe 1/1/3
```

The following example shows the details of the VLANs on which ND inspection is enabled for a VRF.

```
device (config-vrf-3)# show ipv6 neighbor vrf 3 inspection vlan 2
IPv6 Neighbor inspection VLAN 2: Enabled
  Untrusted Ports : ethe 1/1/1 to 1/1/2
  Trusted Ports : ethe 1/1/3
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show ipv6 ospf

Displays OSPFv3 information.

## Syntax

**show ipv6 ospf**

## Modes

User EXEC mode

## Examples

The following example displays sample output from the **show ipv6 ospf** command.

# show ipv6 ospf area

Displays the OSPFv3 area table in a specified format.

## Syntax

**show ipv6 ospf area** [ *A.B.C.D* ] [ *decimal* ]

## Parameters

*A.B.C.D*

Area address in dotted decimal format.

*decimal*

Area address in decimal format.

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf area** command displays the following information:

| Output field | Description |
|---|---|
| Area | The area number. |
| Interface attached to this area | The device interfaces attached to the area. |
| Number of Area scoped LSAs is *N* | Number of LSAs (*N* ) with a scope of the specified area. |
| SPF algorithm executed is *N* | The number of times (*N* ) the OSPF Shortest Path First (SPF) algorithm is executed within the area. |
| SPF last updated | The interval in seconds that the SPF algorithm was last executed within the area. |
| Current SPF node count | The current number of SPF nodes in the area. |
| Router | Number of router LSAs in the area. |
| Network | Number of network LSAs in the area. |
| Indx | The row number of the entry in the routers's OSPF area table. |
| Statistics of Area | The number of the area whose statistics are displayed. |
| Maximum hop count to nodes. | The maximum number of hop counts to an SPF node within the area. |

# Examples

The following example shows sample output from the **show ipv6 ospf area** command when an area is specified.

```
device> show ipv6 ospf area 400
Area 400:
  Authentication: Not Configured
  Active interface(s)attached to this area: None
  Inactive interface(s)attached to this area: ve 20  ve 30
  Number of Area scoped LSAs is 311
  Sum of Area LSAs Checksum is 9e8fff
  Statistics of Area 400:
    SPF algorithm executed 10 times
    SPF last updated: 5920 sec ago
    Current SPF node count: 1
      Router: 1 Network: 0
      Maximum of Hop count to nodes: 0
```

# show ipv6 ospf database

Displays lists of information about different OSPFv3 link-state advertisements (LSAs).

## Syntax

**show ipv6 ospf database** [ **advrtr** *A.B.C.D* | **extensive** | **link-id** *decimal* | **prefix** *ipv6-addr* ]

**show ipv6 ospf database** [ **as-external** | **inter-prefix** | **inter-router** | **intra-prefix** | **link** [ *decimal* ] | **network** | **router** | **type-7** ] [ **advrtr** *A.B.C.D* | **link-id** *decimal* ]

**show ipv6 ospf database scope** { **area** { *A.B.C.D* | *decimal* } | **as** | **link** }

**show ipv6 ospf database summary**

## Parameters

**advrtr** *A.B.C.D*
Displays LSAs by Advertising Router Id in dotted decimal format.

**extensive**
Displays detailed lists of LSA information.

**link-id** *decimal*
Link-state ID that differentiates LSAs. Valid values range from 1 through 4294967295.

**prefix**
Display LSAs that contain a prefix.

*ipv6-addr*
Specifies an IPv6 address.

**as-external**
Displays information about external LSAs.

**inter-prefix**
Displays information about inter area prefix LSAs.

**inter-router**
Displays information about inter area router LSAs.

**intra-prefix**
Displays information about intra area prefix LSAs.

**link** *decimal*
Displays information about the link LSAs.

**network**
Displays information about network LSAs.

**router**
Displays information about router LSAs.

**type-7**
Displays information about the not so stubby area (NSSA) external LSAs.

**scope**
Displays LSA information by LSA scope.

**area**
Displays LSAs by scope within a specified area.

**as**
Displays autonomous system (AS) LSAs by scope.

**link**
Displays link LSAs by scope.

**summary**
Displays LSA summary information.

# Modes

User EXEC mode

# Command Output

The **show ipv6 ospf database** command displays the following information:

| Output field | Description |
|---|---|
| Area ID | The OSPF area in which the device resides. |
| Type | Type of LSA. LSA types can be the following:<br>• Rtr - Router LSAs (Type 1).<br>• Net - Network LSAs (Type 2).<br>• Inap - Inter-area prefix LSAs for ABRs (Type 3).<br>• Inar - Inter-area router LSAs for ASBRs (Type 4).<br>• Extn - AS external LSAs (Type 5).<br>• Link - Link LSAs (Type 8).<br>• Iap - Intra-area prefix LSAs (Type 9). |
| LS ID | The ID of LSA in Decimal. |
| Adv Rtr | The device that advertised the route. |
| Seq(Hex) | The sequence number of the LSA. The OSPF neighbor that sent the LSA stamps it with a sequence number to enable the device and other OSPF routers to determine which LSA for a given route is the most recent. |
| Age | The age of the LSA, in seconds. |
| Chksum | A checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The device uses the checksum to verify that the packet is not corrupted. |
| Len | The length, in bytes, of the LSA. |
| Sync | Sync status with the slave management processor (MP). |

The **show ipv6 ospf database extensive** command displays the following information:

| Output field | Description |
|---|---|
| **Router LSA (Type 1) (Rtr) Fields** | |

| Output field | Description |
|---|---|
| Capability Bits | A bit that indicates the capability of the device. The bit can be set to one of the following:<br>B - The device is an area border router.<br>E - The device is an AS boundary router.<br>V - The device is a virtual link endpoint.<br>W - The device is a wildcard multicast receiver. |
| Options | A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following:<br>V6 - The device should be included in IPv6 routing calculations.<br>E - The device floods AS-external-LSAs as described in RFC 2740.<br>MC - The device forwards multicast packets as described in RFC 1586.<br>N - The device handles type 7 LSAs as described in RFC 1584.<br>R - The originator is an active router.<br>DC -The device handles demand circuits. |
| Type | The type of interface. Possible types can be the following:<br>Point-to-point - A point-to-point connection to another router.<br>Transit - A connection to a transit network.<br>Virtual link - A connection to a virtual link. |
| Metric | The cost of using this router interface for outbound traffic. |
| Interface ID | The ID assigned to the router interface. |
| Neighbor Interface ID | The interface ID that the neighboring router has been advertising in hello packets sent on the attached link. |
| Neighbor Router ID | The router ID (IPv4 address) of the neighboring router that advertised the route. (By default, the router ID is the IPv4 address configured on the lowest numbered loopback interface. If the device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device.) |
| **Network LSA (Type 2) (Net) Fields** | |
| Options | A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following:<br>V6 - The device should be included in IPv6 routing calculations.<br>E - The device floods AS-external-LSAs as described in RFC 2740.<br>MC - The device forwards multicast packets as described in RFC 1586.<br>N - The device handles type 7 LSAs as described in RFC 1584.<br>R - The originator is an active router.<br>DC -The device handles demand circuits. |
| Attached Router | The address of the neighboring router that advertised the route. |
| **Inter-Area Prefix LSA (Type 3) (Inap) Fields** | |
| Metric | The cost of the route. |

| Output field | Description |
|---|---|
| Prefix Options | An 8-bit field describing various capabilities associated with the prefix. |
| Prefix | The IPv6 prefix included in the LSA. |
| **Inter-Area Router LSA (Type 4) (Inar) Fields** | |
| Options | A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following:<br>V6 - The device should be included in IPv6 routing calculations.<br>E - The device floods AS-external-LSAs as described in RFC 2740.<br>MC - The device forwards multicast packets as described in RFC 1586.<br>N - The device handles type 7 LSAs as described in RFC 1584.<br>R - The originator is an active router.<br>DC -The device handles demand circuits. |
| Metric | The cost of the route. |
| Destination Router ID | The ID of the router described in the LSA. |
| **AS External LSA (Type 5) (Extn) Fields** | |
| Bits | The bit can be set to one of the following:<br>• E - If bit E is set, a Type 2 external metric. If bit E is zero, a Type 1 external metric.<br>• F - A forwarding address is included in the LSA.<br>• T - An external route tag is included in the LSA. |
| Metric | The cost of this route, which depends on bit E. |
| Prefix Options | An 8-bit field describing various capabilities associated with the prefix. |
| Referenced LS Type | If non-zero, an LSA with this LS type is associated with the LSA. |
| Prefix | The IPv6 prefix included in the LSA. |
| **Link LSA (Type 8) (Link) Fields** | |
| Router Priority | The router priority of the interface attaching the originating router to the link. |
| Options | The set of options bits that the router would like set in the network LSA that will be originated for the link. |
| Link Local Address | The originating router's link-local interface address on the link. |
| Number of Prefix | The number of IPv6 address prefixes contained in the LSA. |
| Prefix Options | An 8-bit field of capabilities that serve as input to various routing calculations:<br>• NU - The prefix is excluded from IPv6 unicast calculations.<br>• LA - The prefix is an IPv6 interface address of the advertising router.<br>• MC - The prefix is included in IPv6 multicast routing calculations.<br>• P - NSSA area prefixes are readvertised at the NSSA area border. |
| Prefix | The IPv6 prefix included in the LSA. |
| **Intra-Area Prefix LSAs (Type 9) (Iap) Fields** | |

| Output field | Description |
|---|---|
| Number of Prefix | The number of prefixes included in the LSA. |
| Referenced LS Type, Referenced LS ID | Identifies the router-LSA or network-LSA with which the IPv6 address prefixes are associated. |
| Referenced Advertising Router | The address of the neighboring router that advertised the route. |
| Prefix Options | An 8-bit field describing various capabilities associated with the prefix. |
| Metric | The cost of using the advertised prefix. |
| Prefix | The IPv6 prefix included in the LSA. |
| Number of Prefix | The number of prefixes included in the LSA. |

# Examples

The following example shows sample output from the **show ipv6 ospf database** command.

```
device> show ipv6 ospf database

LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID         Type LSID       Adv Rtr         Seq(Hex) Age  Cksum Len   Sync
0.0.0.200       Link 897        192.168.98.213  80000007 1277 9044  64    Yes
0.0.0.200       Link 136        192.168.98.111  80000007 582  fb0b  64    Yes
0.0.0.200       Link 2049       192.168.98.213  80000006 1277 381a  64    Yes
0.0.0.200       Link 1156       192.168.98.111  80000007 582  cf38  64    Yes
0.0.0.200       Link 2052       192.168.98.213  80000004 799  5b06  64    Yes
0.0.0.200       Rtr  0          192.168.98.111  800002ea 823  cb7b  56    Yes
0.0.0.200       Rtr  0          192.168.98.213  800001c7 799  8402  56    Yes
0.0.0.200       Net  1156       192.168.98.111  80000004 823  b2d2  32    Yes
0.0.0.200       Net  136        192.168.98.111  80000008 823  aed2  32    Yes
N/A             Extn 0000021d   10.223.223.223     800000a8 1319 441e  32       Yes
```

The following example shows sample output from the **show ipv6 ospf database** command when the **advr** keyword is used.

```
device> show ipv6 ospf database advr 192.168.98.111

LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID         Type LSID       Adv Rtr         Seq(Hex) Age  Cksum Len   Sync
0.0.0.200       Link 136        192.168.98.111  80000007 634  fb0b  64    Yes
     Router Priority: 1
     Options: V6E---R--
     LinkLocal Address: fe80::768e:f8ff:fe3e:1800
     Number of Prefix: 1
     Prefix Options:
     Prefix: 5100::193:213:111:0/112
```

The following example shows sample output from the **show ipv6 ospf database** command when the **as-external** keyword is used.

```
device> show ipv6 ospf database as-external

LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID         Type LSID       Adv Rtr         Seq(Hex) Age  Cksum Len   Sync
N/A             Extn 2          192.168.98.213  80000004 895  6e5e  44    Yes
     Bits: E--
     Metric: 0
     Prefix Options:
     Referenced LSType: 0
     Prefix: 5100:213:213:0:192:213:1:0/112
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID         Type LSID       Adv Rtr         Seq(Hex) Age  Cksum Len   Sync
N/A             Extn 1          192.168.98.190  80001394 643  1cc9  28    Yes
     Bits: E--
     Metric: 1
     Prefix Options:
     Referenced LSType: 0
     Prefix: ::/0
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID         Type LSID       Adv Rtr         Seq(Hex) Age  Cksum Len   Sync
N/A             Extn 2          192.168.98.71   80000258 132  a3ff  32    Yes
     Bits: E-T
     Metric: 1
     Prefix Options:
     Referenced LSType: 0
     Prefix: ::/0
     Tag: 1
```

The following example shows sample output from the **show ipv6 ospf database** command when the **extensive** keyword is used.

```
device> show ipv6 ospf database extensive

LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LS ID    Adv Rtr         Seq(Hex) Age  Cksum Len  Sync
0.0.0.200    Link 897      192.168.98.213  80000007 1432 9044  64    Yes
     Router Priority: 1
     Options: V6E---R--
     LinkLocal Address: fe80::214:ff:fe77:96ff
     Number of Prefix: 1
     Prefix Options:
     Prefix: 5100::193:213:111:0/112
LSA Key - Rtr:Router Net:Network Inap:InterPrefix Inar:InterRouter
          Extn:ASExternal Grp:GroupMembership Typ7:Type7 Link:Link
          Iap:IntraPrefix Grc:Grace
Area ID      Type LSID     Adv Rtr         Seq(Hex) Age  Cksum Len   Sync
0.0.0.200    Link 136      192.168.98.111  80000007 737  fb0b  64    Yes
     Router Priority: 1
     Options: V6E---R--
     LinkLocal Address: fe80::768e:f8ff:fe3e:1800
--More--, next page: Space, next line: Return key, quit: Control-c
```

# show ipv6 ospf interface

Displays interface information for all or specific OSPFv3-enabled interfaces.

## Syntax

**show ipv6 ospf interface** [ **brief** ] [ **ethernet** *unit/slot/port* ] [ **loopback** *number* ] [ **tunnel** *number* ] [ **ve** *vlan_id* ]

## Parameters

**brief**
Displays brief summary information about OSPFv3-enabled interfaces.

**ethernet** *unit/slot/port*
Specifies the physical interface. On standalone devices as well as stacked devices specifies the interface ID in the format unit/slot/port. On standalone devices, "1" is the unit number.

**loopback** *number*
Specifies a loopback port number in the range of 1 to 255.

**tunnel** *number*
Specifies a tunnel interface.

**ve** *vlan_id*
Specifies the VLAN number.

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **brief** keyword to limit the display to the following fields:

- Interface
- Area
- Status
- Type
- Cost
- State
- Nbrs(F/C)

## Command Output

The **show ipv6 ospf interface** command displays the following information:

| This field | Displays |
|---|---|
| Interface status | The status of the interface. Possible status includes the following:<br>• Up.<br>• Down. |
| Type | The type of OSPFv3 circuit running on the interface. Possible types include the following:<br>• BROADCAST<br>• POINT TO POINT UNKNOWN<br>• POINT TO POINT |
| IPv6 Address | The IPv6 address assigned to the interface. |
| Instance ID | An identifier for an instance of OSPFv3. |
| Router ID | The IPv4 address of the device. By default, the router ID is the IPv4 address configured on the lowest numbered loopback interface. If the device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device. |
| Area ID | The IPv4 address or numerical value of the area in which the interface belongs. |
| Cost | The overhead required to send a packet through the interface. |
| Interface bandwidth | The configured bandwidth on a tunnel interface for routing metric purposes only. |
| default | Shows whether or not the default passive state is set. |
| State | The state of the interface. Possible states include the following:<br>• DR - The interface is functioning as the Designated Router for OSPFv3.<br>• BDR - The interface is functioning as the Backup Designated Router for OSPFv3.<br>• Loopback - The interface is functioning as a loopback interface.<br>• P2P - The interface is functioning as a point-to-point interface.<br>• Passive - The interface is up but it does not take part in forming an adjacency.<br>• Waiting - The interface is trying to determine the identity of the BDR for the network.<br>• None - The interface does not take part in the OSPF interface state machine.<br>• Down - The interface is unusable. No protocol traffic can be sent or received on such a interface.<br>• DR other - The interface is a broadcast or NBMA network on which another router is selected to be the DR.<br>• Active - The interface sends or receives all the OSPFv3 control packets, and forms the adjacency. |
| Transmit delay | The amount of time, in seconds, it takes to transmit Link State Updates packets on the interface. |
| Priority | The priority used when selecting the DR and the BDR. If the priority is 0, the interface does not participate in the DR and BDR election. |
| Timer intervals | The interval, in seconds, of the hello-interval, dead-interval, and retransmit-interval timers. |

| This field | Displays |
|---|---|
| DR | The router ID (IPv4 address) of the DR. |
| BDR | The router ID (IPv4 address) of the BDR. |
| Number of I/F scoped LSAs | The number of interface LSAs scoped for a specified area, AS, or link. |
| DR Election | The number of times the DR election occurred. |
| Delayed LSA Ack | The number of the times the interface sent a delayed LSA acknowledgement. |
| Neighbor Count | The number of neighbors to which the interface is connected. |
| Adjacent Neighbor Count | The number of neighbors with which the interface has formed an active adjacency. |
| Neighbor | The router ID (IPv4 address) of the neighbor. This field also identifies the neighbor as a DR or BDR, if appropriate. |
| Interface statistics | The following statistics are provided for the interface:<br>• Unknown - The number of Unknown packets transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received Unknown packets.<br>• Hello - The number of Hello packets transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received Hello packets.<br>• DbDesc - The number of Database Description packets transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received Database Description packets.<br>• LSReq - The number of link-state requests transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received link-state requests.<br>• LSUpdate - The number of link-state updates transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received link-state requests.<br>• LSAck - The number of link-state acknowledgements transmitted and received by the interface. Also, the total number of bytes associated with transmitted and received link-state acknowledgements. |

# Examples

This example shows sample output from the **show ipv6 ospf interface** command when no arguments or keywords are used.

```
device# show ipv6 ospf interface

e 1/1/9 admin up, oper up, IPv6 enabled
  IPv6 Address:
      fe80::224:10ff:fe76:4bc0
      201::1/64
  Instance ID 0, Router ID 2.2.2.2
  Area ID 0, Cost 1, Type BROADCAST
  MTU: 1500
  State DR, Transmit Delay 1 sec, Priority 1, Link-LSA Tx not suppressed
  Timer intervals :
    Hello 10, Hello Jitter 10  Dead 40, Retransmit 5
  Authentication Use: Enabled
   KeyRolloverTime(sec): Configured: 300 Current: 0
   KeyRolloverState: NotActive
   Outbound: None
   Inbound: None
  DR:2.2.2.2 BDR:1.1.1.1  Number of I/F scoped LSAs is 2
  DRElection:     2 times, DelayedLSAck:   425 times
  Neighbor Count = 1,   Adjacent Neighbor Count= 1
     Neighbor:
      1.1.1.1 (BDR)
    Statistics of interface e 1/1/9:
      Type     tx         rx         tx-byte    rx-byte
      Unknown  0          0          0          0
      Hello    80035      80133      3201392    3205320
      DbDesc   5          3          240        144
      LSReq    1          1          28         76
      LSUpdate 2095       1262       171228     92540
      LSAck    425        419        32020      48604
      OSPF messages dropped,no authentication: 0
```

This example shows sample output from the **show ipv6 ospf interface** command when the **brief** keyword is used.

```
device# show ipv6 ospf interface brief
Interface     Area        Status Type Cost  State    Nbrs(F/C)
e 1/1/9       0           up     BCST 1     DR       1/1
e 1/1/12      0           down   BCST 0     Down     0/0
ve 20         0           up     BCST 1     DR       0/0
ve 60         0           up     BCST 1     DR       0/0
ve 310        0           down   BCST 0     Down     0/0
ve 360        0           down   BCST 0     Down     0/0
loopback 1    0           up     BCST 1     Loopback 0/0
loopback 2    0           up     BCST 1     Loopback 0/0
loopback 3    0           up     BCST 1     Loopback 0/0
```

This example shows information about a specified OSPF-enabled Ethernet interface, including the cost, where the cost is calculated using the default interface speed and auto cost.

```
device# show ipv6 ospf interface ethernet 3/1/1

     e 3/1/1 admin up, oper up, ospf enabled, state up
     fe80::224:10ff:fe76:4bc0
      201::1/64,
     Area 0
     Database Filter: Not Configured
     State BDR, Pri 1, Cost 1, Options 2, Type broadcast Events 3
```

This example shows information about a specified OSPF-enabled Ethernet interface, including the cost, which has been calculated using the configured interface bandwidth and the default auto-cost.

```
device# show ipv6 ospf interface ethernet 3/1/1

     e 1/1/3 admin up, oper up, IPv6 enabled
  IPv6 Address:
      fe80::ce4e:24ff:fe6d:bc00
      9000:1111:9013::2/64
  Instance ID 0, Router ID 192.168.3.1
  Area ID 0, Cost 34, Type BROADCAST
  MTU: 1500
  State DR, Transmit Delay 1 sec, Priority 1, Link-LSA Tx not suppressed
  Timer intervals :
    Hello 10, Hello Jitter 10  Dead 40, Retransmit 5
  Authentication Use: Enabled
   KeyRolloverTime(sec): Configured: 300 Current: 0
   KeyRolloverState: Not Active
   Outbound: None
   Inbound: None
  DR:192.168.3.1 BDR:192.168.1.1  Number of I/F scoped LSAs is 2
  DRElection:     2 times, DelayedLSAck:     1 times
  Neighbor Count = 1,   Adjacent Neighbor Count= 1
     Neighbor:
      192.168.1.1 (BDR)
    Statistics of interface e 1/1/3:
      Type     tx         rx         tx-byte    rx-byte
      Unknown  0          0          0          0
      Hello    82         78         3268       3120
      DbDesc   2          3          116        304
      LSReq    1          1          148        28
      LSUpdate 16         7          1144       1048
      LSAck    1          3          156        328
      OSPF messages dropped, no authentication: 0
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was modified to include configured bandwidth status. |

# show ipv6 ospf memory

Displays information about OSPFv3 memory usage.

## Syntax

**show ipv6 ospf memory**

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf memory** command displays the following information:

| Output field | Description |
|---|---|
| Total Dynamic Memory Allocated | A summary of the amount of dynamic memory allocated, in bytes, to OSPFv3. |
| Memory Type | The type of memory used by OSPFv3. (This information is for use by Ruckus technical support in case of a problem.) |
| Size | The size of a memory type. |
| Allocated | The amount of memory currently allocated to a memory type. |
| Max-alloc | The maximum amount of memory that was allocated to a memory type. |
| Alloc-Fails | The number of times an attempt to allocate memory to a memory type failed. |
| Global memory pool for all instances | A summary of the amount of memory allocated from heap. |

# Examples

The following is sample output from the **show ipv6 ospf memory** command.

```
device> show ipv6 ospf memory

Total Dynamic Memory Allocated for this instance : 4296579 bytes
   Memory Type               Size       Allocated  Max-alloc  Alloc-Fails
   MTYPE_OSPF6_AREA          471191     1          4          0
   MTYPE_OSPF6_AREA_RANGE    29         0          16         0
   MTYPE_OSPF6_SUMMARY_ADDRE 25         0          16         0
   MTYPE_OSPF6_IF            280        1          64         0
   MTYPE_OSPF6_NEIGHBOR      12502      1          32         0
   MTYPE_OSPF6_ROUTE_NODE    21         1          4096       0
   MTYPE_OSPF6_ROUTE_INFO    35         1          4096       0
   MTYPE_OSPF6_PREFIX        20         0          16         0
   MTYPE_OSPF6_LSA           129        3          4096       0
   MTYPE_OSPF6_VERTEX        166        1          64         0
   MTYPE_OSPF6_SPFTREE       44         1          2          0
   MTYPE_OSPF6_NEXTHOP       28         2          256        0
   MTYPE_OSPF6_EXTERNAL_INFO 40         0          4096       0
   MTYPE_THREAD              32         5          1024       0
   MTYPE_OSPF6_LINK_LIST     20         3098       20480      0
   MTYPE_OSPF6_LINK_NODE     12         19         20480      0
   MTYPE_OSPF6_LSA_RETRANSMI 6          3          8192       0
global memory pool for all instances
   Memory Type               Size       Allocated  Max-alloc  Alloc-Fails
   MTYPE_OSPF6_TOP           61475      1          1          0
   MTYPE_OSPF6_LSA_HDR       56         3          4          0
   MTYPE_OSPF6_RMAP_COMPILED 0          0          0          0
   MTYPE_OSPF6_OTHER         0          0          0          0
   MTYPE_THREAD_MASTER       84         1          1          0
```

# show ipv6 ospf neighbor

Displays OSPFv3 neighbor information.

## Syntax

**show ipv6 ospf neighbor** [ **detail** | **router-id** *A.B.C.D* ]

## Parameters

**detail**

Displays detailed neighbor information.

**router-id** *A.B.C.D*

Displays neighbor information for the specified router ID.

## Modes

User EXEC mode

## Command Output

The **show ip ospf neighbor** command displays the following information:

| Output field | Description |
|---|---|
| Router ID | The IPv4 address of the neighbor. By default, the router ID is the IPv4 address configured on the lowest numbered loopback interface. If the device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device. |
| Pri | The OSPFv3 priority of the neighbor. The priority is used during election of the DR and BDR. |
| State | The state between the device and the neighbor. The state can be one of the following:<br>• Down<br>• Attempt<br>• Init<br>• 2-Way<br>• ExStart<br>• Exchange<br>• Loading<br>• Full |
| DR | The router ID (IPv4 address) of the DR. |
| BDR | The router ID (IPv4 address) of the BDR. |
| Interface [State] | The interface through which the router is connected to the neighbor. The state of the interface can be one of the following:<br>• DR - The interface is functioning as the Designated Router for OSPFv3. |

| Output field | Description |
|---|---|
| | • BDR - The interface is functioning as the Backup Designated Router for OSPFv3.<br>• Loopback - The interface is functioning as a loopback interface.<br>• P2P - The interface is functioning as a point-to-point interface.<br>• Passive - The interface is up but it does not take part in forming an adjacency.<br>• Waiting - The interface is trying to determine the identity of the BDR for the network.<br>• None - The interface does not take part in the OSPF interface state machine.<br>• Down - The interface is unusable. No protocol traffic can be sent or received on such an interface.<br>• DR other - The interface is a broadcast or NBMA network on which another router is selected to be the DR. |

The **show ip ospf neighbor router-id** command displays the following information:

| Output field | Description |
|---|---|
| Router ID | The IPv4 address of the neighbor. By default, the router ID is the IPv4 address configured on the lowest numbered loopback interface. If the device does not have a loopback interface, the default router ID is the lowest numbered IPv4 address configured on the device. |
| Pri | The OSPFv3 priority of the neighbor. The priority is used during election of the DR and BDR. |
| State | The state between the device and the neighbor. The state can be one of the following:<br>• Down<br>• Attempt<br>• Init<br>• 2-Way<br>• ExStart<br>• Exchange<br>• Loading<br>• Full |
| DR | The router ID (IPv4 address) of the DR. |
| BDR | The router ID (IPv4 address) of the BDR. |
| Interface [State] | The interface through which the router is connected to the neighbor. The state of the interface can be one of the following:<br>• DR - The interface is functioning as the Designated Router for OSPFv3.<br>• BDR - The interface is functioning as the Backup Designated Router for OSPFv3.<br>• Loopback - The interface is functioning as a loopback interface.<br>• P2P - The interface is functioning as a point-to-point interface. |

| Output field | Description |
|---|---|
| | • Passive - The interface is up but it does not take part in forming an adjacency.<br>• Waiting - The interface is trying to determine the identity of the BDR for the network.<br>• None - The interface does not take part in the OSPF interface state machine.<br>• Down - The interface is unusable. No protocol traffic can be sent or received on such a interface.<br>• DR other - The interface is a broadcast or NBMA network on which another router is selected to be the DR. |
| DbDesc bit | The Database Description packet, which includes 3 bits of information:<br>• The first bit can be "i" or "-". "i" indicates the inet bit is set. "-" indicates the inet bit is not set.<br>• The second bit can be "m" or "-". "m" indicates the more bit is set. "-" indicates the more bit is not set.<br>• The third bit can be "m" or "s". An "m" indicates the master. An "s" indicates standby. |
| Index | The ID of the LSA from which the neighbor learned of the router. |
| DR Decision | The router ID (IPv4 address) of the neighbor's elected DR and BDR. |
| Last Received Db Desc | The content of the last database description received from the specified neighbor. |
| Number of LSAs in Db Desc retransmitting | The number of LSAs that need to be retransmitted to the specified neighbor. |
| Number of LSAs in Summary List | The number of LSAs in the neighbor's summary list. |
| Number of LSAs in Request List | The number of LSAs in the neighbor's request list. |
| Number of LSAs in Retransmit List | The number of LSAs in the neighbor's retransmit list. |
| Seqnum Mismatch | The number of times sequence number mismatches occurred. |
| BadLSReq | The number of times the neighbor received a bad link-state request from the device. |
| One way received | The number of times a hello packet, which does not mention the router, is received from the neighbor. This omission in the hello packet indicates that the communication with the neighbor is not bidirectional. |
| Inactivity Timer | The number of times that the neighbor's inactivity timer expired. |
| Db Desc Retransmission | The number of times sequence number mismatches occurred. |
| LSReqRetrans | The number of times the neighbor retransmitted link-state requests to the device. |
| LSUpdateRetrans | The number of times the neighbor retransmitted link-state updates to the device. |
| LSA Received | The number of times the neighbor received LSAs from the device. |
| LS Update Received | The number of times the neighbor received link-state updates from the device. |

# Examples

The following is sample output from the **show ipv6 ospf neighbor** command.

```
device> show ipv6 ospf neighbor

Total number of neighbors in all states: 2
Number of neighbors in state Full      : 2
RouterID        Pri State    DR              BDR             Interface  [State]
192.168.98.111    1 Full     192.168.98.111  192.168.98.213  e 4/3/1    [BDR]
192.168.98.111    1 Full     192.168.98.111  192.168.98.213  ve 17      [BDR]
```

The following is sample output from the **show ipv6 ospf neighbor** command when the **router-id** keyword is sued.

```
device> show ipv6 ospf neighbor router-id 1192.168.98.111

RouterID        Pri State    DR              BDR             Interface     [State]
192.168.98.111    1 Full     192.168.98.111  192.168.98.213  e 4/3/1       [BDR]
                Option: 00-00-13    QCount: 0    Timer: 73
    DbDesc bit for this neighbor: --m
    Nbr Ifindex of this router: 136
    Nbr DRDecision: DR 192.168.98.111, BDR 192.168.98.213
    Last received DbDesc: opt:xxx ifmtu:0 bit:--s seqnum:0
    Number of LSAs in DbDesc retransmitting: 0
    Number of LSAs in SummaryList: 0
    Number of LSAs in RequestList: 0
    Number of LSAs in RetransList: 0
    SeqnumMismatch       0 times, BadLSReq             0 times
    OnewayReceived       0 times, InactivityTimer      0 times
    DbDescRetrans        0 times, LSReqRetrans         0 times
    LSUpdateRetrans     11 times
    LSAReceived        379 times, LSUpdateReceived   258 times
RouterID        Pri State    DR              BDR             Interface     [State]
192.168.98.111    1 Full     192.168.98.111  192.168.98.213  ve 17         [BDR]
                Option: 00-00-13    QCount: 0    Timer: 44
    DbDesc bit for this neighbor: --m
    Nbr Ifindex of this router: 1156
    Nbr DRDecision: DR 192.168.98.111, BDR 192.168.98.213
    Last received DbDesc: opt:xxx ifmtu:0 bit:--s seqnum:0
    Number of LSAs in DbDesc retransmitting: 0
    Number of LSAs in SummaryList: 0
    Number of LSAs in RequestList: 0
    Number of LSAs in RetransList: 0
    SeqnumMismatch       0 times, BadLSReq             0 times
    OnewayReceived       0 times, InactivityTimer      0 times
    DbDescRetrans        0 times, LSReqRetrans         0 times
    LSUpdateRetrans      3 times
    LSAReceived        317 times, LSUpdateReceived   262 times
```

# show ipv6 ospf redistribute route

Displays all IPv6 routes or a specified IPv6 route that the device has redistributed into OSPFv3.

## Syntax

**show ipv6 ospf redistribute route** *A.B.C.D:M*

## Parameters

*A.B.C.D:M*

Specifies an IPv6 network prefix.

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf redistribute route** command displays the following information:

| Output field | Description |
|---|---|
| ID | An ID for the redistributed route. |
| Prefix | The IPv6 routes redistributed into OSPFv3. |
| Protocol | The protocol from which the route is redistributed into OSPFv3. Redistributed protocols can be the following:<br>• BGP - BGP4+.<br>• RIP - RIPng.<br>• Static - IPv6 static route table.<br>• Connected - A directly connected network. |
| Metric Type | The metric type used for routes redistributed into OSPFv3. The metric type can be the following:<br>• Type-1 - Specifies a small metric (2 bytes).<br>• Type-2 - Specifies a big metric (3 bytes). |
| Metric | The value of the default redistribution metric, which is the OSPF cost of redistributing the route into OSPFv3. |

## Examples

The following is sample output from the **show ipv6 ospf redistribute route** command when no IPv6 network prefix is specified.

```
device> show ipv6 ospf redistribute route

Id      Prefix                                  Protocol  Metric Type  Metric
1       5100::192:213:163:0/112                 Connect   Type-2       0
2       5100:213:213:0:192:213:1:0/112          Connect   Type-2       0
```

The following is sample output from the **show ipv6 ospf redistribute route** command when an IPv6 network prefix is specified.

```
device> show ipv6 ospf redistribute route 2001:db8::
Id     Prefix                          Protocol  Metric Type  Metric
1      2001:db8::/32                   Static    Type-2       1
```

# show ipv6 ospf routes

Displays OSPFv3 routes.

## Syntax

**show ipv6 ospf routes** *A.B.C.D:M*

## Parameters

*A.B.C.D:M*

Specifies a destination IPv6 address.

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf routes** command displays the following information:

| Output field | Description |
|---|---|
| Current Route Count (Displays with the entire OSPFv3 route table only) | The number of route entries currently in the OSPFv3 route table. |
| Intra/Inter/External (Type1/Type2) (Displays with the entire OSPFv3 route table only) | The breakdown of the current route entries into the following route types:<br>• Inter - The number of routes that pass into another area.<br>• Intra - The number of routes that are within the local area.<br>• External1 - The number of type 1 external routes.<br>• External2 - The number of type 2 external routes. |
| Equal-cost multi-path (Displays with the entire OSPFv3 route table only) | The number of equal-cost routes to the same destination in the OSPFv3 route table. If load sharing is enabled, the device equally distributes traffic among the routes. |
| Destination | The IPv6 prefixes of destination networks to which the device can forward IPv6 packets. "*IA" indicates the next router is an intra-area router. |
| Cost | The type 1 cost of this route. |
| E2 Cost | The type 2 cost of this route. |
| Tag | The route tag for this route. |
| Flags | Flags associated with this route. |
| Dis | Administrative Distance for this route. |
| Next-Hop Router | The IPv6 address of the next router a packet must traverse to reach a destination. |
| Outgoing Interface | The router interface through which a packet must traverse to reach the next-hop router. |
| Adv_Router | The IP address of the advertising router. |

# Examples

The following example displays the entire OSPFv3 route table for the device.

```
device> show ipv6 ospf routes

Current Route count: 309
    Intra: 304 Inter: 4 External: 1 (Type1 0/Type2 1)
    Equal-cost multi-path: 56
    OSPF Type: IA- Intra, OA - Inter, E1 - External Type1, E2 - External Type2
   Destination                    Cost       E2Cost     Tag        Flags    Dis
E2 ::/0                           2          1          0          00000003 110
   Next_Hop_Router                Outgoing_Interface Adv_Router
   fe80::768e:f8ff:fe3e:1800      e 4/3/1            192.168.98.111
   fe80::768e:f8ff:fe3e:1800      ve 17              192.168.98.111
   Destination                    Cost       E2Cost     Tag        Flags    Dis
IA 5100::192:61:1001:0/112        3          0          0          00000007 110
   Next_Hop_Router                Outgoing_Interface Adv_Router
   fe80::768e:f8ff:fe3e:1800      e 4/3/1            192.168.98.111
   fe80::768e:f8ff:fe3e:1800      ve 17              192.168.98.111
   Destination                    Cost       E2Cost     Tag        Flags    Dis
IA 5100::192:111:2:111/128        1          0          0          00000007 110
   Next_Hop_Router                Outgoing_Interface Adv_Router
   fe80::768e:f8ff:fe3e:1800      e 4/3/1            192.168.98.111
   fe80::768e:f8ff:fe3e:1800      ve 17              192.168.98.111
   Destination                    Cost       E2Cost     Tag        Flags    Dis
IA 5100::192:111:3:111/128        1          0          0          00000007 110
   Next_Hop_Router                Outgoing_Interface Adv_Router
   fe80::768e:f8ff:fe3e:1800      e 4/3/1            192.168.98.111
--More--, next page: Space, next line: Return key, quit: Control-c
```

# show ipv6 ospf spf

Displays OSPFv3 SPF node, table, and tree information.

## Syntax

**show ipv6 ospf spf** { **node** | **table** | **tree** } [ **area** { *A.B.C.D* | *decimal* } ]

## Parameters

**node**

Displays OSPFv3 node information.

**table**

Specifies a SPF table.

**tree**

Specifies a SPF tree.

**area**

Specifies an area.

*A.B.C.D*

Area address in dotted decimal format.

*decimal*

Area address in decimal format.

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf spf node** command displays the following information:

| Output field | Description |
|---|---|
| SPF node | Each SPF node is identified by its device ID (IPv4 address). If the node is a child node, it is additionally identified by an interface on which the node can be reached appended to the router ID in the format *router-id* :*interface-id* . |
| Cost | The cost of traversing the SPF node to reach the destination. |
| Hops | The number of hops needed to reach the parent SPF node. |
| Next Hops to Node | The IPv6 address of the next hop-router or the router interface through which to access the next-hop router. |
| Parent Nodes | The SPF node's parent nodes. A parent node is an SPF node at the highest level of the SPF tree, which is identified by its router ID. |
| Child Nodes | The SPF node's child nodes. A child node is an SPF node at a lower level of the SPF tree, which is identified by its router ID and interface on which the node can be reached. |

The **show ipv6 ospf spf table** command displays the following information:

| Output field | Description |
|---|---|
| Destination | The destination of a route, which is identified by the following:<br>• "R", which indicates the destination is a router. "N", which indicates the destination is a network.<br>• An SPF node's device ID (IPv4 address). If the node is a child node, it is additionally identified by an interface on which the node can be reached appended to the router ID in the format *router-id* :*interface-id* . |
| Bits | A bit that indicates the capability of the device. The bit can be set to one of the following:<br>• B - The device is an area border router.<br>• E - The device is an AS boundary router.<br>• V - The device is a virtual link endpoint.<br>• W - The device is a wildcard multicast receiver. |
| Options | A 24-bit field that enables IPv6 OSPF routers to support the optional capabilities. When set, the following bits indicate the following:<br>V6 - The router should be included in IPv6 routing calculations.<br>E - The router floods AS-external-LSAs as described in RFC 2740.<br>MC - The router forwards multicast packets as described in RFC 1586.<br>N - The router handles type 7 LSAs as described in RFC 1584.<br>R - The originator is an active router.<br>DC -The router handles demand circuits. |
| Cost | The cost of traversing the SPF node to reach the destination. |
| Next hop | The IPv6 address of the next hop-router. |
| Interface | The router interface through which to access the next-hop router. |

# Examples

The following example shows information about SPF nodes.

```
device> show ipv6 ospf spf node

SPF node for Area 0.0.0.200
 SPF node 192.168.98.213,  cost: 0,  hops: 0
  nexthops to node:
  parent nodes:
  child nodes: 192.168.98.111:136 192.168.98.111:1156
 SPF node 192.168.98.111:136,  cost: 1,  hops: 1
  nexthops to node:    :: e 4/3/1
  parent nodes: 192.168.98.213
  child nodes: 192.168.98.111:0
 SPF node 192.168.98.111:1156,  cost: 1,  hops: 1
  nexthops to node:    :: ve 17
  parent nodes: 192.168.98.213
  child nodes: 192.168.98.111:0
 SPF node 192.168.98.111:0,  cost: 1,  hops: 2
  nexthops to node:    fe80::768e:f8ff:fe3e:1800 e 4/3/1
                       fe80::768e:f8ff:fe3e:1800 ve 17
  parent nodes: 192.168.98.111:136 192.168.98.111:1156
  child nodes:
SPF node for Area 400
 SPF node 192.168.98.213,  cost: 0,  hops: 0
  nexthops to node:
  parent nodes:
  child nodes:
SPF node for Area 0.0.0.0
 SPF node 192.168.98.213,  cost: 0,  hops: 0
  nexthops to node:
  parent nodes:
  child nodes: 192.168.98.111:0
 SPF node 192.168.98.111:0,  cost: 1,  hops: 1
  nexthops to node:    5100::192:113:111:111 VLink 1
  parent nodes: 192.168.98.213
  child nodes: 192.168.98.61:5 192.168.98.190:1551 192.168.98.112:643
 SPF node 192.168.98.61:5,  cost: 2,  hops: 2
  nexthops to node:    5100::192:113:111:111 VLink 1
  parent nodes: 192.168.98.111:0
  child nodes: 192.168.98.61:0
 SPF node 192.168.98.190:1551,  cost: 2,  hops: 2
  nexthops to node:    5100::192:113:111:111 VLink 1
--More--, next page: Space, next line: Return key,
```

The following example shows information about SPF nodes in area 0.

```
device> show ipv6 ospf spf node area 0

SPF node for Area 0
SPF node 10.223.223.223,  cost: 0,  hops: 0
 nexthops to node:
 parent nodes:
 child nodes: 10.223.223.223:88
SPF node 10.223.223.223:88,  cost: 1,  hops: 1
 nexthops to node:    :: ethe 1/3/2
 parent nodes: 10.223.223.223
 child nodes: 10.1.1.1:0
SPF node 10.1.1.1:0,  cost: 1,  hops: 2
 nexthops to node:    fe80::2e0:52ff:fe91:bb37 ethe 1/3/2
 parent nodes: 10.223.223.223:88
 child nodes:
```

The following example displays the SPF table for area 0.

```
device> show ipv6 ospf spf table area 0

SPF table for Area 0.0.0.200
  Destination          Bits Options  Cost  Nexthop                   Interface
R 192.168.98.111      --V-B V6E---R-   1  fe80::768e:f8ff:fe3e:1800 e 4/3/1
R 192.168.98.111      --V-B V6E---R-   1  fe80::768e:f8ff:fe3e:1800 ve 17
N 192.168.98.111[136] ----- V6E---R-   1  ::                        e 4/3/1
N 192.168.98.111[1156 ----- V6E---R-   1  ::                        ve 17
 SPF table for Area 400
  Destination           Bits Options  Cost  Nexthop                   Interface
 SPF table for Area 0.0.0.0
  Destination           Bits Options  Cost  Nexthop                   Interface
R 192.168.98.71       ---E- V6E---RD   4  fe80::768e:f8ff:fe3e:1800 e 4/3/1
R 192.168.98.71       ---E- V6E---RD   4  fe80::768e:f8ff:fe3e:1800 ve 17
R 192.168.98.190      ---E- V6E---R-   2  fe80::768e:f8ff:fe3e:1800 e 4/3/1
R 192.168.98.190      ---E- V6E---R-   2  fe80::768e:f8ff:fe3e:1800 ve 17
```

The following example displays the SPF tree for area 0.

```
device> show ipv6 ospf spf tree area 0

   SPF tree for Area 0
 +- 10.223.223.223 cost 0
     +- 10.223.223.223:88 cost 1
         +- 10.1.1.1:0 cost 1
```

# show ipv6 ospf summary

Displays summary information for all OSPFv3 instances.

## Syntax

**show ipv6 ospf summary**

## Modes

User EXEC mode

## Examples

```
device> show ipv6 ospf summary

Seq Instance        Intfs   Nbrs    Nbrs-Full LSAs    Routes
1   default-vrf     5       2       1         12      2
```

# show ipv6 ospf virtual-links

Displays information about all OSPFv3 virtual links or specified links.

## Syntax

**show ipv6 ospf virtual-links** [ **brief** ]

## Parameters

**brief**

Displays summary information.

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf virtual-links** command displays the following information:

| Output field | Description |
|---|---|
| Index | An index number associated with the virtual link. |
| Transit Area ID | The ID of the shared area of two ABRs that serves as a connection point between the two routers. |
| Router ID | Router ID of the router at the other end of the virtual link (virtual neighbor). |
| Interface Address | The local address used to communicate with the virtual neighbor. |
| State | The state of the virtual link. Possible states include the following:<br>• P2P - The link is functioning as a point-to-point interface.<br>• DOWN - The link is down. |

# Examples

The following is sample output from the **show ipv6 ospf virtual-links** command when no arguments or keywords are used:

```
device> show ipv6 ospf virtual-link

Transit Area ID  Router ID        Interface Address             State
0.0.0.200        192.168.98.111   5100::192:213:111:213         P2P
  Timer intervals(sec) :
    Hello 10, Hello Jitter 10,  Dead 40, Retransmit 5, TransmitDelay 1
  DelayedLSAck:    65 times
  Authentication: Not Configured
  Statistics:
      Type     tx         rx         tx-byte    rx-byte
      Unknown  0          0          0          0
      Hello    819        816        32760      32640
      DbDesc   10         11         300        11008
      LSReq    6          0          6492       0
      LSUpdate 1579       1161       138284     101488
      LSAck    65         52         29340      29532
      OSPF messages dropped,no authentication: 0
  Neighbor: State: Full Address: 5100::192:113:111:111 Interface: e 4/3/1
```

# show ipv6 ospf virtual-neighbor

Displays information about OSPFv3 virtual neighbors.

## Syntax

**show ipv6 ospf virtual-neighbor** [ **brief** ]

## Parameters

**brief**

Displays summary information.

## Modes

User EXEC mode

## Command Output

The **show ipv6 ospf virtual-neighbor** command displays the following information:

| Output field | Description |
|---|---|
| Index | An index number associated with the virtual neighbor. |
| Router ID | IPv4 address of the virtual neighbor. |
| Address | The IPv6 address to be used for communication with the virtual neighbor. |
| State | The state between the device and the virtual neighbor. The state can be one of the following:<br>• Down<br>• Attempt<br>• Init<br>• 2-Way<br>• ExStart<br>• Exchange<br>• Loading<br>• Full |
| Interface | The interface type. |
| Option | The bits set in the virtual-link hello or database descriptors. |
| QCount | The number of packets that are in the queue and ready for transmission. If the system is stable, this number should always be 0. |
| Timer | A timer that counts down until a hello packet should arrive. If "timers" elapses and a hello packet has not arrived, the VL neighbor is declared to be down. |

# Examples

The following is sample output from the **show ipv6 ospf virtual-neighbor** command when no arguments or keywords are used:

```
device> show ipv6 ospf virtual-neighbor

Index Router ID        Address                       State     Interface
1     10.14.14.14      2001:db8:44:44::4             Full      eth 1/1/8
                Option: 00-00-00    QCount: 0    Timer: 408
2     10.14.14.14      2001:db8:44:44::4             Full      tunnel 256
                Option: 00-00-00    QCount: 0    Timer: 43
```

# show ipv6 pim anycast-rp

Displays information for an IPv6 PIM Anycast rendezvous point (RP) interface.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **anycast-rp**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim anycast-rp** command displays the following information:

| Output Field | Description |
|---|---|
| Number of Anycast RP | Specifies the number of Anycast RP sets in the multicast domain. |
| Anycast RP | Specifies a shared RP address used among multiple PIM routers. |
| ACL ID | Specifies the ACL ID assigned. |
| ACL Name | Specifies the name of the Anycast RP set. |
| ACL Filter | Specifies the ACL filter state SET or UNSET. |
| Peer List | Specifies host addresses that are permitted in the Anycast RP set. |

## Examples

The following example shows information for an IPv6 PIM Anycast RP interface.

```
device> show ipv6 pim anycast-rp
Number of Anycast RP: 1
Anycast RP: 1001::1
  ACL ID: 200
  ACL Name: my-anycast-rp-set
  ACL Filter: SET
  Peer List:
   1:1:1::1
   2:2:2::2
   3:3:3::3
```

# show ipv6 pim bsr

Displays information on a device that has been elected as the bootstrap router (BSR).

## Syntax

**show ipv6 pim** [ **all-vrf** | **vrf** *vrf-name* ] **bsr**

## Parameters

**all-vrf**

Displays information for all VRF instances.

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim bsr** command displays the following information:

| Output Field | Description |
|---|---|
| BSR address | The IPv6 address of the interface configured as the IPv6 PIM Sparse (BSR. |
| BSR priority | The priority assigned to the interface for use during the BSR election process. During BSR election, the priorities of the candidate BSRs are compared and the interface with the highest BSR priority becomes the BSR. |
| Hash mask length | The number of significant bits in the IPv6 multicast group comparison mask. This mask determines the IPv6 multicast group numbers for which the device can be a BSR. The default is 32 bits, which allows the device to be a BSR for any valid IPv6 multicast group number.<br><br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| Next bootstrap message in | Indicates how many seconds will pass before the BSR sends its next Bootstrap message.<br><br>**NOTE**<br>This field appears only if this device is the BSR. |
| Next Candidate-RP-advertisement message in | Indicates how many seconds will pass before the BSR sends its next candidate RP advertisement message.<br><br>**NOTE**<br>This field appears only if this device is a candidate BSR. |

| Output Field | Description |
|---|---|
| RP | Indicates the IPv6 address of the Rendezvous Point (RP).<br><br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| group prefixes | Indicates the multicast groups for which the RP listed by the previous field is a candidate RP.<br><br>**NOTE**<br>This field appears only if this device is a candidate BSR. |
| Candidate-RP-advertisement period | Indicates how frequently the BSR sends candidate RP advertisement messages.<br><br>**NOTE**<br>This field appears only if this device is a candidate BSR. |

# Examples

The following example shows information for a device that has been elected as the BSR.

```
device> show ipv6 pim bsr
PIMv2 Bootstrap information for Vrf Instance : default-vrf
------------------------------------------------------------------
This system is the Elected BSR
BSR address: 2006:1001::1. Hash Mask Length 64. Priority 32.
Next bootstrap message in 00:01:00
Configuration:
 Candidate loopback 1 (Address 2006:1001::1). Hash Mask Length 64. Priority 32.
Next Candidate-RP-advertisment in 00:00:50
 RP: 2006:1001::1
  group prefixes:
  ff00:: / 8
 Candidate-RP-advertisement period: 60
 Candidate-RP-advertisement period: 60

 Candidate-RP-advertisement period: 60
```

The following example shows information for a device that is not the BSR.

```
device> show ipv6 pim bsr
PIMv2 Bootstrap information for Vrf Instance : default-vrf
------------------------------------------------------------
BSR address: 2006:1001::1. Hash Mask Length 64. Priority 32.
This system is not a Candidate-RP.
This system is not a Candidate-RP.
```

# show ipv6 pim counter

Displays the number of default VLAN ID changes that occurred and how many times a tagged port was placed in a VLAN since the applicable VRF was created.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **counter**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim counter** command displays the following information:

| Output Field | Description |
|---|---|
| DFTVlanChange | The number of default VLAN ID changes that have occurred since the applicable VRF was created. |
| VlanPort | The number of times that a tagged port was placed in a VLAN since the applicable VRF was created. |

## Examples

The following example displays default-VLAN-ID change and tagged-port information.

```
device> show ipv6 pim vrf eng counter
Event Callback:
 DFTVlanChange : 0    VlanPort : 0
LP to MP IPCs:
 SM_REGISTER : 8315   MCAST_CREATE :     0
 S_G_AGEOUT :     3   WRONG_IF :         0
 ABOVE_THRESHOLD: 0   MCAST_FIRST_DATA : 3
 SET KAT :        3   SET KAT INFINITY : 3
MP to LP IPCs:
 INIT : 25            INSERT_VPORT :  30
 DELETE_VPORT : 186   DELETE_VIF :   162
 MOVE_VPORT :     0   DEL_ENTRY :     16
 INSERT_SOURCE :  0   DELETE_SOURCE :  0
 RESET_SRC_LIST : 0   MOVE_TNNL_PORT : 0
 FLAG_CHANGE :    6   FDB_VIDX_CHANGE: 0
 OIF_FLAG_CHANGE :0
Error Counters:
 PIM_PKT_DRP :  0     PIM_PKT_DRP(Glb) : 0
 MCGRP_PKT_DRP: 0     MCGRP_PKT_DRP(Gl): 0
 RPSET_MAXED :  0
```

# show ipv6 pim group

Displays IPv6 PIM group information.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **group**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim group** command displays the following information:

| Output Field | Description |
|---|---|
| Total number of Groups | Lists the total number of IPv6 multicast groups the device is forwarding. |
| Group | The multicast group address. |
| Group member at | Interface name and number. |

## Examples

The following example displays IPv6 PIM group information.

```
device# show ipv6 pim group
Total number of groups: 1
1    Group ff7e:a40:2001:3e8:27:0:1:2
     Group member at  e 1/3/1: v31
```

# show ipv6 pim hw-resource

Displays usage and fail-count information for SG entries on virtual routing and forwarding instances (VRFs).

## Syntax

**show ipv6 pim** [ **all-vrf** | **vrf** *vrf-name* ] **hw-resource**

## Parameters

**all-vrf**

Displays hardware resource information for all VRFs.

**vrf** *vrf-name*

Specifies displaying hardware resource information for a particular VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim hw-resource** command displays the following information:

| Output Field | Description |
|---|---|
| VRF | Name of the VRF. |
| Usage | Number of allocated SG entries in this VRF. |
| Fail | Number of failures while allocating SG entries in this VRF (due to system-max limit). |
| Total usage | Total number of SG entries in the system (All-VRFs). |
| System-max limit for SG entries | Configured system limit using the **pim6-hw-mcache** command. |

## Examples

The following example displays hardware resource information for all VRFs.

```
device> show ipv6 pim all-vrf hw-resource
              VRF  In-Use     Fail
       default-vrf   3072        8
              blue   3072        0
   ------------------------------
       Total usage   6144

   System-max limit for SG entries: 6144
```

# show ipv6 pim interface

Displays information for IPv6 PIM interfaces.

## Syntax

**show ipv6 pim interface** { **ethernet** *unit* **/** *slot* **/** *port* | **loopback** *loopback-number* | **ve** *ve-number* }

## Parameters

**ethernet***unit* **/***slot* **/** *port*
Specifies a physical interface. On standalone devices, use "1" as the unit number.

**loopback** *loopback-number*
Specifies a loopback interface.

**ve** *ve-number*
Specifies a virtual interface.

## Modes

Privileged EXEC mode

## Examples

The following example displays output from the **show ipv6 pim interface** command, showing that ACL f10 is applied to interface 1/1/9 to control neighbor access.

```
Device# show ipv6 pim interface
Flags     : SM  - Sparse Mode v2

--------+---------------+-----+---+-------+---+-----+-------+-------+----+--------+

Int'face|Local          |Mode |St |Des Rtr|TTL|Mcast| Filter| VRF   |DR  |Override
        |Address        |     |   |Add Prt|Thr|Bndry|  ACL  |       |Prio|Interval
--------+---------------+-----+---+-------+---+-----+-------+-------+----+--------+
e1/1/1   3000::2         SM   Ena  Itself   1  None   None   default  1   3000ms
e1/1/9   201::1          SM   Ena  Itself   1  None   f10    default  1   3000ms
e1/1/12  1222::1         SM   Dis  Itself   1  None   None   default  1   3000ms
v20      2000::2         SM   Ena  Itself   1  None   None   default  1   3000ms
v60      6000::1         SM   Ena  Itself   1  None   None   default  1   3000ms
v310     1100::2         SM   Dis  Itself   1  None   None   default  1   3000ms
v360     1600::1         SM   Dis  Itself   1  None   None   default  1   3000ms
l2       4444::2         SM   Ena  Itself   1  None   None   default  1   3000ms
l3       7711::11        SM   Ena  Itself   1  None   None   default  1   3000ms
Total Number of Interfaces : 9
```

## History

| Release version | Command history |
|---|---|
| 08.0.20a | This command was modified to display neighbor filter information. |

# show ipv6 pim mcache

Displays the IPv6 PIM multicast cache.

## Syntax

**show ipv6 pim** [**vrf** *vrf-name* ] **mcache** [ **counts** ] [ *source-address group-address* | **dit-idx** *dit-idx* | **g_entries** | **receiver** { **ethernet** *stack/slot/port* | **vlan** *vlan-num* } | **sg_entries** | **sparse** | **ssm** ]

## Parameters

**vrf** *vrf-name*
Specifies IPv6 PIM multicast cache information for a VRF instance.

*source-address*
Specifies the multicast cache source address.

*group-address*
Specifies the multicast cache group address.

**counts**
Specifies the number of entries.

**dense**
Displays only the PIM Dense Mode entries.

**dit-idx** *dit-idx*
Displays all entries that match a specified directory information tree (DIT).

**g_entries**
Displays only the (*, G) entries.

**receiver**
Displays all entries that egress a specified interface.

**ethernet** *stack/slot/port*
Specifies the Ethernet interface which is the receiver.

**vlan** *vlan-num*
Specifies the VLAN which is the receiver.

**sg_entries**
Displays only the (S, G) entries.

**sparse**
Displays only the PIM Sparse Mode entries.

**ssm**
Displays only the SSM entries.

## Modes

User EXEC mode

# Command Output

The **show ipv6 pim mcache** command displays the following information:

| Field | Description |
|---|---|
| Total entries in mcache | Shows the total number of IPv6 PIM mcache entries. |
| upstream neighbor | Shows the upstream neighbor for the Source/RP based on the type of entry.<br>For (*,G) it shows the upstream neighbor towards the RP. For (S,G) entries it<br>shows the upstream neighbor towards the source. |
| Flags | Show the flags associated with the forward entry. |
| slow ports ethe | Shows the forwarding port ID of the mcache entry which is in the software forwarding path. |
| AgeSltMsk | Shows the slot number on which MP expects ingress traffic. |
| L2 FID | Shows the hardware resource allocated for the traffic switched to receivers in the ingress VLAN. |
| DIT | Shows the hardware resource allocated for routed receivers. |
| Forwarding_oif | Shows the number of outgoing interfaces of the mcache entry. |
| immediate_oifs | Shows the local immediate outgoing interface of the mcache entry. |
| blocked_oifs | Shows the PIM Sparse mode blocked outgoing interfaces. |
| L3 (SW) 1 | Shows whether the traffic is switched or routed out of the interface. |
| L3 (HW) 1 | The forwarding entries by using hardware. |
| Src-Vlan | VLAN associated with the ingress interface. |

# Examples

The following example displays the IPv6 PIM multicast cache.

```
device> show ipv6 pim mcache
IP Multicast Mcache Table
Entry Flags    : SM  - Sparse Mode, SSM - Source Specific Multicast, DM - Dense Mode
                 RPT    - RPT Bit, SPT - SPT Bit, LSRC - Local Source, LRCV - Local Receiver
                 HW - HW Forwarding Enabled, FAST - Resource Allocated, TAG - Need For Replication Entry
                 REGPROB - Register In Progress, REGSUPP - Register Suppression Timer
                 MSDPADV - Advertise MSDP, NEEDRTE - Route Required for Src/RP,  PRUN - DM Prune
Upstream
Interface Flags: IM - Immediate, IH - Inherited, WA - Won Assert
                 MJ - Membership Join, MI - Membership Include, ME - Membership Exclude
                 BR - Blocked RPT, BA - Blocked Assert, BF - Blocked Filter, BI - Blocked IIF
Total entries in mcache: 4
1    (*, ff05::4422) RP 2006:1001::1, in v503 (tag e2/1/11), Uptime 1d 00:27:26 (SM)
     upstream neighbor fe80::204:ff:fe05:6 (2006:503::1001)
     Flags (0x002604a2) SM RPT LRCV TAG
     slow ports: ethe 3/1/1
     AgeSltMsk: 0, L2 FID: 8192, DIT: NotReq, profile: none
     Forwarding_oif: 1, Immediate_oif: 1, Blocked_oif: 0
     L3 (SW) 1:
       e3/1/1(VL170), 1d 00:27:26/0, Flags: MJ
2    (2006:170::1010, ff34::500) in v170 (tag e3/1/1), Uptime 00:37:51, Rate 0 (SM)
     Source is directly connected. RP 2006:1001::1
     Flags (0x20429ce1) SM SPT REG L2REG LSRC HW FAST TAG
     fast ports: ethe 2/1/11
     AgeSltMsk: 1, L2 FID: 4188, DIT: 1
     Forwarding_oif: 1, Immediate_oif: 1, Blocked_oif: 0
     L3 (HW) 1:
       TR(e2/1/11,e2/1/11)(VL503), 00:37:26/183, Flags: IM
    Src-Vlan: 170
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | The output of the command was modified to remove the AvgRate and Profile entries. |

# show ipv6 pim resource

Displays the hardware resource information, such as hardware allocation, availability, and limit, for software data structures.

## Syntax

**show ipv6 pim** [ **all-vrf** | **vrf** *vrf-name* ] **resource**

## Parameters

**all-vrf**

Displays information for all virtual routing and forwarding instances (VRFs).

**vrf** *vrf-name*

Displays information for a particular VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim resource** command displays the following information:

| Output Field | Description |
|---|---|
| Num alloc | Number of allocated PIM resources. |
| System max | Maximum number of VRF resources. |
| Size | Internal size. |
| alloc | Number of nodes of that data that are currently allocated in memory. |
| in-use | Number of allocated nodes in use. |
| avail | Number of allocated nodes are not in use. |
| get-fail | Number of allocated notes that failed. |
| limit | Maximum number of nodes that can be allocated for a data structure. This may or may not be configurable, depending on the data structure |
| get-mem | Current memory allocation. |
| size | Unit size. |
| init | Initial number. |

# Examples

The following example displays output from the **show ipv6 pim resource** command.

```
device> show ipv6 pim vrf white res
Global PIM Parameters :-
GLOBAL Ipv6 MULTICAST CLASS Size:23573 bytes
GLOBAL Ipv6 PIM CLASS Size:2162 bytes
MULTICAST IPV6 CLASS Num alloc:2, System max:17, Size:1346 bytes
PIM IPV6 CLASS Num alloc:2, System max:17, Size:50485
Vrf Instance : white
-------------------------------------
                   alloc in-use avail get-fail limit  get-mem  size init
NBR list              64      2    62        0    512       73    96   64
RP set list          256      1   255        0   1536    12824    49  256
Static RP             64      0    64        0     64        0    42   64
LIF Entry            512      0   512        0    512        0    47  512
Anycast RP            64      0    64        0     64        0   190   64
timer                 64      0    64        0  14848       65    64   64
prune                 32      0    32        0   7424        0    34   32
pimsm J/P elem      1024      0  1024        0  48960   640448    29  128
Timer Data           512      2   510        0  14848     1409    28   64
mcache SLIB Sync    1120      2  1118        0  64960     9502    34  280
mcache               896      2   894        0  12992     5570 1144   56
graft if no mcache   197      0   197        0  45704        0    64  197
HW replic vlan      1000      2   998        0 116000   170179    66  500
HW replic port      1024      2  1022        0  59392   170179    81  256
pim/dvm intf. group   64      0    64        0  14848        0    24   64
pim/dvm global group 512      0   512        0  14848     6700    46   64
repl entry(Global)  1024      2  1022        0 237568    40644    49 1024
MLD Resources(All Vrfs):
groups              1024      0  1024        0   4096     7100   328  256
phy-ports           2048      0  2048        0   4096     7600   148  256
exist-phy-port      1792      0  1792        0  12992   196484    62   56
group-query           56      0    56        0  12992        0    84   56
Hardware-related Resources:
Total (S,G) entries 2
Total SW FWD entries 0
 Total sw w/Tag MVID entries 0
 Total sw w/Tag invalid MVID entries 0
Total HW FWD entries 2
 Total hw w/Tag MVID entries 2
 Total hw w/Tag invalid MVID entries 0
```

# show ipv6 pim rp-candidate

Displays candidate rendezvous point (RP) information.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **rp-candidate**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim rp-candidate** command displays the following information:

| Field | Description |
|---|---|
| Candidate-RP-advertisement in | Indicates how many seconds will pass before the BSR sends its next RP message.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |
| RP | Indicates the IPv6 address of the RP.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |
| group prefixes | Indicates the multicast groups for which the RP listed by the previous field is a candidate RP.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |
| Candidate-RP-advertisement period | Indicates how frequently the BSR sends candidate RP advertisement messages.<br><br>**NOTE**<br>This field appears only if this device is a candidate RP. |

# Examples

The following example shows information for a candidate RP.

```
device> show ipv6 pim rp-candidate
Next Candidate-RP-advertisement in 00:00:10
  RP: 1be::11:21
    group prefixes:
    ff00:: / 8
  Candidate-RP-advertisement period: 60
```

# show ipv6 pim rpf

Displays what PIM sees as the best reverse path to the source. While there may be multiple routes back to the source, the one displayed by this command is the one that PIM thinks is best.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **rpf** *ipv6-address* [ *group-address* ]

## Parameters

**vrf** *vrf-name*
Displays information for a VRF instance.

*ipv6-address*
Specifies the source IPv6 address for reverse-path forwarding (RPF) check.

*group-address*
Specifies the group IPv6 address for reverse-path forwarding (RPF) check.

## Modes

User EXEC mode

## Examples

The following example shows best reverse path to the specified source.

```
device> show ipv6 pim rpf 2008:165::1010
upstream nbr 2006:503::1001 on v503
```

# show ipv6 pim rp-hash

Displays rendezvous-point (RP) information for an IPv6 PIM Sparse group.

## Syntax

**show ipv6 pim** [**vrf** *vrf-name* ] **rp-hash** *group-addr*

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

*group-addr*

Specifies the address of an IPv6 PIM Sparse IP multicast group.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim rp-hash** command displays the following information:

| Output Field | Description |
|---|---|
| RP | Indicates the IPv6 address of the RP for the specified IPv6 PIM Sparse group. Following the IPv6 address is the port or virtual interface through which this device learned the identity of the RP. |
| Info source | Indicates the IPv6 address on which the RP information was received. Following the IPv6 address is the method through which this device learned the identity of the RP. |

## Examples

The following example shows RP information for an IPv6 PIM Sparse group.

```
device# show ipv6 pim rp-hash ff1e::1:2
  RP: 2001:3e8:255:255::17, v2
    Info source: 2001:3e8:255:255::17, via bootstrap
```

# show ipv6 pim rp-map

Displays rendezvous-point (RP)-to-group mapping information.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **rp-map**

## Parameters

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim rp-map** command displays the following information:

| Output Field | Description |
|---|---|
| Index | The index number of the table entry in the display. |
| Group address | Indicates the IPv6 PIM Sparse multicast group address using the listed RP. |
| RP address | Indicates the Iv6 address of the RP for the listed PIM Sparse group. |

## Examples

The following example shows RP-to-group mapping.

```
device #show ipv6 pim rp-map
Number of group-to-RP mappings: 3
--------------------------------------------------------------------------------
S.No  Group address   RP address
--------------------------------------------------------------------------------
1     ff07::c:1       3200:12::32
2     ff07::c:2       3200:12::32
3     ff07::c:3       3200:12::32
Number of group-to-RP mappings: 3
```

# show ipv6 pim rp-set

Displays rendezvous-point (RP)-set list for the device elected as the bootstrap router (BSR).

## Syntax

**show ipv6 pim** [ **all-vrf** | **vrf** *vrf-name* ] **rp-set**

## Parameters

**all-vrf**

Displays information for all VRF instances.

**vrf** *vrf-name*

Displays information for the specified VRF instance.

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim rp-set** command displays the following information:

| Output Field | Description |
|---|---|
| Number of group prefixes | The number of IPv6 PIM Sparse group prefixes for which the RP is responsible. |
| Group prefix | Indicates the multicast groups for which the RP listed by the previous field is a candidate RP. |
| RPs expected or received | Indicates how many RPs were expected and received in the latest Bootstrap message. |
| RP *num* | Indicates the RP number. If there are multiple RPs in the IPv6 PIM Sparse domain, a line of information for each of them is listed, and they are numbered in ascending numerical order. |
| priority | The RP priority of the candidate RP. During the election process, the candidate RP with the highest priority is elected as the RP. |
| age | The age (in seconds) of this RP-set.<br>**NOTE**<br>If this device is not a BSR, this field contains zero. Only the BSR ages the RP-set. |

# Examples

The following example shows the RP set list.

```
device> show ipv6 pim rp-set
Static RP
---------
Static RP count: 1
100::1
Number of group prefixes Learnt from BSR: 0
No RP-Set present
```

# show ipv6 pim sparse

Displays PIM Sparse configuration information for IPv6, including whether the hardware-drop feature is enabled or disabled, information for PIM SSM range ACL configuration, and route-precedence settings.

## Syntax

**show ipv6 pim** [ **vrf** *vrf-name* ] **sparse**

## Parameters

**vrf** *vrf-name*

Displays IPv6 PIM information for a virtual routing and forwarding instance (VRF).

## Modes

User EXEC mode

## Command Output

The **show ipv6 pim sparse** command displays the following information:

| Output Field | Displays |
|---|---|
| **Global PIM Sparse mode settings** | |
| Maximum mcache | Maximum number of multicast cache entries. |
| Current Count | Number of multicast cache entries used. |
| Hello interval | How frequently the device sends IPv6 PIM Sparse hello messages to its IPv6 PIM Sparse neighbors. This field shows the number of seconds between hello messages. IPv6 PIM Sparse routers use hello messages to discover one another. |
| Neighbor timeout | Number of seconds the device waits for a hello message from a neighbor before determining that the neighbor is no longer present and is not removing cached IPv6 PIM Sparse forwarding entries for the neighbor. Default is 105 seconds. |
| Join or Prune interval | How frequently the device sends IPv6 PIM Sparse Join or Prune messages for the multicast groups it is forwarding. This field shows the number of seconds between Join or Prune messages.<br>The device sends Join or Prune messages on behalf of multicast receivers that want to join or leave an IPv6 PIM Sparse group. When forwarding packets from IPv6 PIM Sparse sources, the device sends the packets only on the interfaces on which it has received join requests in Join or Prune messages for the source group. |
| Inactivity interval | Number of seconds a forwarding entry can remain unused before the router deletes it. Default is 180 sec. |
| Hardware Drop Enabled | Indicates whether hardware drop is enabled or disabled.<br>To prevent unwanted multicast traffic from being sent to the CPU, PIM Routing and Passive Multicast Route Insertion (PMRI) can be used together to ensure that multicast streams are only |

| Output Field | Displays |
|---|---|
| | forwarded out ports with interested receivers and unwanted traffic is dropped in the hardware on Layer 3 Switches. |
| Prune Wait Interval | Number of seconds a PIM device waits before stopping traffic to neighbor devices that do not want the traffic. Range is from zero to three seconds. Default is three seconds. |
| Bootstrap Msg interval | How frequently the BSR configured on the device sends the RP set to the RPs within the IPv6 PIM Sparse domain. The RP set is a list of candidate RPs and their group prefixes. The group prefix of a candidate RP indicates the range of IPv6 PIM Sparse group numbers for which it can be an RP.<br><br>**NOTE**<br>This field contains a value only if an interface on the device is elected to be the BSR. Otherwise, the field is blank. |
| Candidate-RP Msg interval | Number of seconds the candidate RP configured on the Layer 3 switch sends candidate RP advertisement messages to the BSR. Default is 60 seconds. |
| Register Suppress Time | This is the mean interval between receiving a Register-Stop and allowing<br><br>registers to be sent again. A lower value means more frequent register bursts at RP, while a higher value means longer join latency for new receivers. Default: 60 seconds. |
| Register Probe Time | Number of seconds the PIM router waits for a register-stop from an RP before it generates another NULL register to the PIM RP. Default is 10 seconds. |
| Register Stop Delay | Register stop message. Default is 10 seconds. |
| Register Suppress interval | Number of seconds that it takes the designated router to send Register-encapsulated data to the RP after receiving a Register-Stop message. Default is 60 seconds. |
| SSM Enabled | If yes, source-specific multicast is configured globally on this router. |
| SPT threshold | Number of packets the device sends using the path through the RP before switching to the SPT path. Default is 1 packet. |
| SSM Group Range | Source-specific multicast group range. |
| Route Precedence | The route precedence configured to control the selection of routes based on the four route types:<br>• Non-default route from the mRTM<br>• Default route from the mRTM<br>• Non-default route from the uRTM<br>• Default route from the uRTM |
| Embedded RP Enabled | Indicates whether the embedded RP is enabled or disabled. |

# Examples

The following example shows whether the hardware-drop feature has been enabled or disabled.

```
device> show ipv6 pim sparse

Global PIM Sparse Mode Settings
  Maximum Mcache          : 4096        Current Count              : 7
  Hello interval          : 30          Neighbor timeout           : 105
  Join/Prune interval     : 60          Inactivity interval        : 180
  Hardware Drop Enabled   : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval  : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time  : 60          Register Probe Time        : 10
  Register Stop Delay     : 10          Register Suppress interval : 60
  SSM Enabled             : Yes         SPT Threshold              : 1
  SSM Group Range         : ff3x::/32
  Route Precedence        : mc-non-default mc-default uc-non-default uc-default
  Embedded RP Enabled     : Yes
```

The following example shows IPv6 PIM Sparse configuration information.

```
device> show ipv6 pim sparse

Global PIM Sparse Mode Settings
  Maximum Mcache          : 4096        Current Count              : 7
  Hello interval          : 30          Neighbor timeout           : 105
  Join/Prune interval     : 60          Inactivity interval        : 180
  Hardware Drop Enabled   : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval  : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time  : 60          Register Probe Time        : 10
  Register Stop Delay     : 10          Register Suppress interval : 60
  SSM Enabled             : Yes         SPT Threshold              : 1
  SSM Group Range         : ff3x::/32
  Route Precedence        : mc-non-default mc-default uc-non-default uc-default
  Embedded RP Enabled     : Yes
```

The following examples show the route precedence settings depending on the route-precedence configuration:

```
device(config-ipv6-pim-router)# route-precedence mc-non-default mc-default uc-non-default uc-default
device(config-ipv6-pim-router)# show ipv6 pim sparse

Global PIM Sparse Mode Settings
  Maximum Mcache          : 12992       Current Count              : 2
  Hello interval          : 30          Neighbor timeout           : 105
  Join/Prune interval     : 60          Inactivity interval        : 180
  Hardware Drop Enabled   : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval  : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time  : 60          Register Probe Time        : 10
  Register Stop Delay     : 10          Register Suppress interval : 60
  SSM Enabled             : No          SPT Threshold              : 1
  Route Precedence        : mc-non-default mc-default uc-non-default uc-default
  Embedded RP Enabled     : Yes

device(config-ipv6-pim-router)# route-precedence admin-distance
device(config-ipv6-pim-router)# show ipv6 pim sparse

Global PIM Sparse Mode Settings
  Maximum Mcache          : 12992       Current Count              : 2
  Hello interval          : 30          Neighbor timeout           : 105
  Join/Prune interval     : 60          Inactivity interval        : 180
  Hardware Drop Enabled   : Yes         Prune Wait Interval        : 3
  Bootstrap Msg interval  : 60          Candidate-RP Msg interval  : 60
  Register Suppress Time  : 60          Register Probe Time        : 10
  Register Stop Delay     : 10          Register Suppress interval : 60
  SSM Enabled             : No          SPT Threshold              : 1
  Route Precedence        : admin-distance
  Embedded RP Enabled     : Yes
device(config-ipv6-pim-router)
```

# show ipv6 pim traffic

Displays IPv6 PIM traffic statistics.

## Syntax

**show ipv6 pim traffic** [ **vrf** *vrf-name* ] [ **join-prune** ] [ **rx** | **tx** ]

## Parameters

**vrf** *vrf-name*
Specifies information for a VRF instance.

**join-prune**
Specifies displaying join and prune statistics.

**rx**
Specifies displaying received PIM traffic statistics.

**tx**
Specifies displaying transmitted PIM traffic statistics.

## Modes

Privileged EXEC mode

## Usage Guidelines

PIM control packet statistics for interfaces that are configured for standard PIM are listed first by the display.

## Command Output

The **show ipv6 pim traffic** command displays the following information:

| Output Field | Description |
|---|---|
| Port | The port or virtual interface on which the IPv6 PIM interface is configured. |
| HELLO | The number of IPv6 PIM Hello messages sent or received on the interface. |
| JOIN-PRUNE | The number of Join or Prune messages sent or received on the interface.<br>**NOTE**<br>Unlike PIM dense, PIM Sparse uses the same messages for Joins and Prunes. |
| ASSERT | The number of Assert messages sent or received on the interface. |
| REGISTER GRAFT (DM) | The number of Register messages sent or received on the interface. |

| Output Field | Description |
|---|---|
| REGISTER STOP (SM) | The number of Register Stop messages sent or received on the interface. |
| BOOTSTRAP MSGS (SM) | The number of bootstrap messages sent or received on the interface. |
| CAND. RP ADV. (SM) | The total number of Candidate-RP-Advertisement messages sent or received on the interface.<br>Register Graft (DM) |
| Err | The total number of MLD messages discarded, including a separate counter for those that failed the checksum comparison. |

# Examples

This example shows PIM traffic statistics:

```
Device# show ipv6 pim traffic
Port    HELLO     JOIN-PRUNE  ASSERT    REGISTER  REGISTER  BOOTSTRAP CAND. RP  Err
                                       GRAFT(DM) STOP(SM)  MSGS  (SM) ADV. (SM)
-------+---------+-----------+---------+---------+---------+---------+---------+---
       Rx        Rx          Rx        Rx        Rx        Rx        Rx        Rx
------+---------+-----------+---------+---------+---------+---------+---------+---
v170    0         0           0         0         0         0         0         0
v501    0         0           0         0         0         0         0         0
v503    3302      2524        0         0         0         0         0         0
Port    HELLO     JOIN-PRUNE  ASSERT    REGISTER  REGISTER  BOOTSTRAP CAND. RP  Err
                                       GRAFT(DM) STOP(SM)  MSGS  (SM) ADV. (SM)
-------+---------+-----------+---------+---------+---------+---------+---------+---
       Tx        Tx          Tx        Tx        Tx        Tx        Tx
------+---------+-----------+---------+---------+---------+---------+---------+---
v170    3576      0           0         0         0         0         0
v501    1456      0           0         0         0         0         0
v503    1456      1314        0         0         0         2         0
```

This example shows the number of received IPv6 PIM Hello packets dropped on interface 1/1/9 to because an ACL to control neighbor access is configured on it.

```
Device#show ipv6 pim traffic rx
Port    HELLO  JN-PRN  ASSERT  REG      REG       BTSTRP    CAND RP  Err
                               GRFT(DM) STOP(SM)  MSGS(SM)  ADV.(SM)

-------+--------+------+------+------+------+---------+----------+-----
        Rx       Rx      Rx     Rx     Rx        Rx        Rx
Rx
-------+-------+------+------+-------+------+---------+---------+------
e1/1/1  0       0        0      0      0         0         0       0
e1/1/9  924     0        0      0      0         5         0       914
e1/1/12 0       0        0      0      0         0         0       0
v20     0       0        0      0      0         0         0       0
v60     0       0        0      0      0         0         0       0
v310    0       0        0      0      0         0         0       0
v360    0       0        0      0      0         0         0       0
```

# History

| Release version | Command history |
|---|---|
| 8.0.20a | This command was modified to display, in the Err column, received Hello packets dropped on an interface because of an ACL to control neighbor access. |

# show ipv6 pimsm-snooping cache

Displays the downstream PIM join/prune information for both source-path tree (SPT) and rendezvous-point tree (RPT).

## Syntax

**show ipv6 pimsm-snooping cache** [ **vlan** *vlan-id* ] *ipv6-address* [ **resources** ]

## Parameters

*ipv6-address*
Specifies the IP address.

**vlan** *vlan-id*
Specifies snooping for a VLAN.

**resources**
Specifies PIM SM snooping resources.

## Modes

Privileged exec mode

## Command Output

The **show ipv6 pimsm-snooping cache** command displays the following information:

| Output field | Description |
|---|---|
| SG | (s,g) downstream fsm state for SPT. |
| G | (*,g) downstream fsm state for RPT |

The **show ipv6 pimsm-snooping cache** command displays the following information only when multi-chassis trunking (MCT) is enabled on the VLAN:

| Output field | Description |
|---|---|
| CCEP | Cluster-client-edge port |
| CEP | Cluster-edge port |
| Remote/Local | Join/Prune received on MCT peer or local |

# Examples

The following example shows PIM SM information.

```
Device#show ipv6 pimsm-snooping cache
 OIF Info:
TR - OIF Belongs to Trunk/LAG, Primary port is displayed
SG - (s,g) downstream fsm state:
G - (*,g) downstream fsm state:
     NI : No Info, J : Join, PP : Prune Pending, CLEAN : cleanup in progress
RPT - (s,g,rpt) downstream fsm state:
     NI : No Info, P : Pruned, PP : Prune Pending, Px : Temp step in (*,G)
     join processing, PPx : Temp State in (*,G) processing, CLEAN : cleanup
     in progress.

PIMSM Snoop cache for vlan 503
1    (* ff7e::1:2:3) Up Time: 03:43:40
                OIF: 1
                TR(e1/1/4) G : J(183) ET: 210, Up Time: 03:43:40

2    (3000::10 ff7e::1:2:3) Up Time: 00:02:52
                OIF: 1
                TR(e1/1/4) SG : J(185) ET: 210, Up Time: 00:02:52
```

The following example shows PIM SM information for a VLAN.

```
Device#show ipv6 pimsm-snooping vlan 503
OIF Info:
TR - OIF Belongs to Trunk/LAG, Primary port is displayed
SG - (s,g) downstream fsm state:
G - (*,g) downstream fsm state:
     NI : No Info, J : Join, PP : Prune Pending, CLEAN : cleanup in progress
RPT - (s,g,rpt) downstream fsm state:
     NI : No Info, P : Pruned, PP : Prune Pending, Px : Temp step in (*,G)
     join processing, PPx : Temp State in (*,G) processing, CLEAN : cleanup
     in progress.

PIMSM Snoop cache for vlan 503
1    (* ff7e::1:2:3) Up Time: 03:43:46
                OIF: 1
                TR(e1/1/4) G : J(177) ET: 210, Up Time: 03:43:46

2    (3000::10 ff7e::1:2:3) Up Time: 00:02:58
                OIF: 1
                TR(e1/1/4) SG : J(179) ET: 210, Up Time: 00:02:58
```

The following example shows PIM SM resource information.

```
Device#show ipv6 pimsm-snooping resources
                      alloc in-use  avail get-fail    limit  get-mem  size init
pimsm group entry      1000      1    999        0   232000        2    64 1000
pimsm source entry     2000      1   1999        0   464000        2    68 2000
pimsm oif entry        2000      1   1999        0   464000        2    89 2000

Total memory in used: 378000 bytes
```

# show ipv6 raguard

Displays the Router Advertisement (RA) guard configuration details.

## Syntax

**show ipv6 raguard** { **counts** | **policy** } { *name* | **all** }

**show ipv6 raguard whitelist** { *number* | **all** }

## Parameters

**counts**

Displays the RA guard permit or drop counts.

**policy**

Displays the RA guard policy details.

**whitelist**

Displays the RA guard whitelist associated with the RA guard policy.

*name*

An ASCII string indicating the name of the RA guard policy, when used along with **counts** keyword, displays the permit or drop counts for the specified RA guard policy. When used with **policy** keyword, displays the configuration of the specified RA guard policy.

**all**

When used with **counts**, **policy**, and **whitelist** keywords, displays the permit or drop counts for all the RA guard policies, configuration of all RA guard policies, and all the associated RA guard whitelists respectively.

*number*

Displays the specific whitelist based on the ID number.

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

The **show ipv6 raguard counts** command is applicable only when logging is enabled on the policy.

# Examples

The following example shows the RA guard drop or permit counts for all RA guard policies:

```
device#show ipv6 raguard counts all
POLICY: policy1
DROPPED-host port: 1
DROPPED-whitelist: 4
DROPPED-prefixlist: 1
DROPPED-max pref: 3
PASSED-trusted port: 0
PASSED-untrusted port: 0
POLICY: policy2
DROPPED-host port: 1
DROPPED-whitelist: 0
DROPPED-prefixlist: 3
DROPPED-max pref: 1
PASSED-trusted port: 0
PASSED-untrusted port: 0
```

The following example shows the details of a RA guard policy p1:

```
device#show ipv6 raguard policy p1
policy:p1
        whitelist:1
```

The following example shows all RA guard whitelist:

```
device#show ipv6 raguard whitelist all
whitelist #1 : 3 entries
        permit fe80:db8::db8:10/128
        permit fe80:db8::db8:5/128
        permit fe80:db8::db8:12/128
```

# show ipv6 rip

Shows RIPng configuration information for the device.

## Syntax

**show ipv6 rip**

## Modes

Privileged EXEC mode or any configuration mode

## Command Output

The **show ipv6 rip** command displays the following information:

| Output field | Description |
|---|---|
| IPv6 RIP status/port | The status of RIPng on the device. Possible status is "enabled" or "disabled." The UDP port number over which RIPng is enabled. |
| Administrative distance | The setting of the administrative distance for RIPng. |
| Updates/expiration | The settings of the RIPng update and timeout timers. |
| Holddown/garbage collection | The settings of the RIPng hold-down and garbage-collection timers. |
| Split horizon/poison reverse | The status of the RIPng split horizon and poison reverse features. Possible status for each is "on" or "off." |
| Default routes | The status of RIPng default routes. |
| Periodic updates/trigger updates | The number of periodic updates and triggered updates sent by the RIPng device. |
| Distribution lists | The inbound and outbound distribution lists applied to RIPng. |
| Redistribution | The types of IPv6 routes redistributed into RIPng. The following types of routes can be redistributed:<br>STATIC<br>CONNECTED<br>BGP - BGP4+<br>OSPF - OSPFv3 |

## Examples

The following example shows settings for RIPng, which is enabled on UDP port 521. Connected, static, OSPFv3, and BGP4+ routes are redistributed through IPv6.

```
device# show ipv6 rip
IPv6 rip enabled, port 521
Administrative distance is 120
Updates every 30 seconds, expire after 180
Holddown lasts 180 seconds, garbage collect after 120
Split horizon is on; poison reverse is off
Default routes are not generated
Periodic updates 5022, trigger updates 10
Distribute List, Inbound : Not set
Distribute List, Outbound
Redistribute: CONNECTED STATIC OSPF BGP
```

# show ipv6 rip route

Displays the RIPng routing table.

## Syntax

**show ipv6 rip route** [ *ipv6-prefix/prefix-length* | *ipv6-address* ]

## Parameters

*ipv6-prefix/prefix-length*

Restricts the display to the entries for the specified IPv6 prefix. You must specify the ipv6-prefix parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. You must specify the prefix-length parameter as a decimal value. A slash mark (/) must follow the ipv6-prefix parameter and precede the prefix-length parameter.

*ipv6-address*

Restricts the display to the entries for the specified IPv6 address. You must specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

## Modes

Privileged EXEC mode or any configuration mode

## Command Output

The **show ipv6 rip route** command displays the following information:

| Output field | Description |
|---|---|
| IPv6 RIP Routing Table entries | The total number of entries in the RIPng routing table. |
| ipv6-prefix /prefix-length<br>ipv6-address | The IPv6 prefix and prefix length.<br>The IPv6 address. |
| Next-hop router | The next-hop router for this device. If :: appears, the route is originated locally. |
| Interface | The interface name. If "null" appears, the interface is originated locally. |
| Source of route | The source of the route information. The source can be one of the following:<br>RIP - routes learned by RIPng.<br>CONNECTED - IPv6 routes redistributed from directly connected networks.<br>STATIC - IPv6 static routes are redistributed into RIPng.<br>BGP - BGP4+ routes are redistributed into RIPng.<br>OSPF - OSPFv3 routes are redistributed into RIPng. |
| Metric number | The cost of the route. The number parameter indicates the number of hops to the destination. |
| Tag number | The tag value of the route. |
| Timers | Indicates if the hold-down (aging) timer or the garbage-collection timer is set. |

# Examples

The following example shows information for a routing table with four entries.

```
device# show ipv6 rip route
IPv6 RIP Routing Table - 4 entries:
ada::1:1:1:2/128, from fe80::224:38ff:fe8f:3000, e 1/3/4
RIP, metric 2, tag 0, timers: aging 17
2001:db8::/64, from fe80::224:38ff:fe8f:3000, e 1/3/4
RIP, metric 3, tag 0, timers: aging 17
bebe::1:1:1:4/128, from ::, null (0)
CONNECTED, metric 1, tag 0, timers: none
cccc::1:1:1:3/128, from fe80::768e:f8ff:fe94:2da, e 2/1/23
RIP, metric 2, tag 0, timers: aging 50
```

# show ipv6 route

To display the IPv6 route table information, use the **show ipv6 route** command.

## Syntax

**show ipv6 route** [ **vrf** *vrf-nam* ] [ *ipv6-address* | *ipv6-prefix/prefix-length* | **bgp** | **connect** | **ospf** | **rip** | **static** | **summary** ]

## Parameters

**vrf** *vrf-name*
Displays the IPv6 route table information for the specified Virtual Routing/Forwarding (VRF) instance.

*ipv6-address*
Restricts the display to the entries for the specified IPv6 address. Specify this parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373.

*ipv6-prefix/prefix-length*
Restricts the display to the entries for the specified IPv6 prefix. Specify the *ipv6-prefix* parameter in hexadecimal using 16-bit values between colons as documented in RFC 2373. Specify the *prefix-length* parameter as a decimal value. A slash mark (/) must follow the *ipv6-prefix* parameter and precede the *prefix-length* parameter.

**bgp**
Displays BGP routes.

**connect**
Displays directly attached routes.

**ospf**
Displays OSPF routes.

**rip**
Displays RIP routes.

**static**
Displays static IPv6 routes.

**summary**
Displays a summary of the prefixes and different route types.

## Modes

Privileged EXEC mode

## Command Output

The **show ipv6 route** command displays the following information:

| Output field | Description |
|---|---|
| Number of entries | The number of entries in the IPv6 route table. |

| Output field | Description |
|---|---|
| Type | The route type, which can be one of the following:<br>• C - The destination is directly connected to the router.<br>• S - The route is a static route.<br>• R - The route is learned from RIPng.<br>• O - The route is learned from OSPFv3.<br>• B - The route is learned from BGP4. |
| IPv6 Prefix | The destination network of the route. |
| Next-Hop Router | The next-hop router. |
| Interface | The interface through which this router sends packets to reach the route's destination. |
| Dis/Metric | The route's administrative distance and metric value. |

The show ipv6 route summary command displays the following information:

| Output field | Description |
|---|---|
| Number of entries | The number of entries in the IPv6 route table. |
| Number of route types | The number of entries for each route type. |
| Number of prefixes | A summary of prefixes in the IPv6 route table, sorted by prefix length. |

# Examples

This example shows how to display the IPv6 route table.

```
device#show ipv6 route
IPv6 Routing Table - 1 entries:
Type Codes - B:BGP C:Connected L:Local O:OSPF R:RIP S:Static
BGP  Codes - i:iBGP e:eBGP
OSPF Codes - i:Inter Area 1:External Type 1 2:External Type 2
STATIC Codes - d:DHCPv6
Type IPv6 Prefix           Next Hop Router    Interface     Dis/Metric     Uptim
e
C    2001:db8::/122        ::                 loopback 11   0/0            14d7h
```

This example shows how to display a summary of the IPv6 route table.

```
device# show ipv6 route summary
IPv6 Routing Table - 7 entries:
   4 connected, 2 static, 0 RIP, 1 OSPF, 0 BGP
   Number of prefixes:
   /16: 1 /32: 1 /64: 3 /128: 2
```

# show ipv6 router

Displays information about the IPv6 routers connected to an IPv6 host.

## Syntax

**show ipv6 router**

## Modes

User EXEC mode

## Usage Guidelines

TheRuckus ICX device can function as an IPv6 host, instead of an IPv6 router, if you configure IPv6 addresses on its interfaces but do not enable IPv6 routing using the ipv6 unicast-routing command. From the IPv6 host, you can display information about IPv6 routers to which the host is connected. The host learns about the routers through their router advertisement messages.

If you configure your device to function as an IPv6 router (you configure IPv6 addresses on its interfaces and enable IPv6 routing using the ipv6 unicast-routing command) and then enter the**show ipv6 router** command, you will get a message that there are no IPv6 router in the table.

## Command Output

The **show ipv6 router** command displays the following information:

| Output field | Description |
|---|---|
| Router IPv6 address on interface port | The IPv6 address for a particular router interface. |
| Last update | The amount of elapsed time (in minutes) between the current and previous updates received from a router. |
| Hops | The default value that should be included in the Hop Count field of the IPv6 header for outgoing IPv6 packets. The hops value applies to the router for which you are displaying information and should be followed by IPv6 hosts attached to the router. A value of 0 indicates that the router leaves this field unspecified. |
| Lifetime | The amount of time (in seconds) that the router is useful as the default router. |
| Reachable time | The amount of time (in milliseconds) that a router assumes a neighbor is reachable after receiving a reachability confirmation. The reachable time value applies to the router for which you are displaying information and should be followed by IPv6 hosts attached to the router. A value of 0 indicates that the router leaves this field unspecified. |
| Retransmit time | The amount of time (in milliseconds) between retransmissions of neighbor solicitation messages. The retransmit time value applies to the router for which you are displaying information and should be followed by IPv6 hosts attached to the router. A value of 0 indicates that the router leaves this field unspecified. |

# Examples

The following example displays information about the IPv6 routers connected to an IPv6 host.

```
device# show ipv6 router
Router fe80::2e0:80ff:fe46:3431 on Ethernet 50, last update 0 min
Hops 64, Lifetime 1800 sec
Reachable time 0 msec, Retransmit time 0 msec
```

# show ipv6 static mroute

Displays information for configured IPv6 multicast routes.

## Syntax

**show ipv6 static mroute** [ **vrf** *vrf-name* | *ipv6-address-prefix/prefix-length* ]

## Parameters

**vrf** *vrf-name*
Specifies a VRF route.

*ipv6-address-prefix/prefix-length*
Specifies an IPv6 address.

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

Only resolved and best static mroutes are added to the mRTM table. These routes are prefixed with an asterisk in the output from the **show ipv6 static mroute** command.

## Examples

Thie following example displays information for configured IPv6 multicast routes:

```
Device(config)# show ipv6 static mroute
IPv6 Static Routing Table - 1 entries:
 IPv6 Prefix                    Interface  Next Hop Router      Met/Dis/Tag Name
*1:1::1:0/120                   ve 90      ::                   1/1/0
```

## History

| Release version | Command history |
|---|---|
| 8.0.10a | This command was introduced. |

# show ipv6 tcp connections

Displays general information about each TCP connection on the router, including the percentage of free memory for each of the internal TCP buffers.

## Syntax

**show ipv6 tcp connections** [ *port-num* | *ipv6-address* ]

## Parameters

*port-num*

Displays the information for the specific port number. Values are SSH:22 TELNET:23 HTTP:80 BGP:179 SSL:443 MSDP:639 LDP:646.

*ipv6-address*

Displays information for the specified IPv6 address of the remote device.

## Modes

User EXEC mode

## Command Output

The **show ipv6 tcp connections** command displays the following information:

| Output field | Description |
|---|---|
| Local IP address:port | The IPv4 or IPv6 address and port number of the local router interface over which the TCP connection occurs. |
| Remote IP address:port | The IPv4 or IPv6 address and port number of the remote router interface over which the TCP connection occurs. |
| TCP state | The state of the TCP connection. Possible states include the following:<br>• LISTEN - Waiting for a connection request.<br>• SYN-SENT - Waiting for a matching connection request after having sent a connection request.<br>• SYN-RECEIVED - Waiting for a confirming connection request acknowledgment after having both received and sent a connection request.<br>• ESTABLISHED - Data can be sent and received over the connection. This is the normal operational state of the connection.<br>• FIN-WAIT-1 - Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent.<br>• FIN-WAIT-2 - Waiting for a connection termination request from the remote TCP.<br>• CLOSE-WAIT - Waiting for a connection termination request from the local user.<br>• CLOSING - Waiting for a connection termination request acknowledgment from the remote TCP.<br>• LAST-ACK - Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request). |

| Output field | Description |
|---|---|
| | • TIME-WAIT - Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request.<br>• CLOSED - There is no connection state. |
| FREE TCP = percentage | The percentage of free TCP control block (TCP) space. |
| FREE TCP QUEUE BUFFER = percentage | The percentage of free TCP queue buffer space. |
| FREE TCP SEND BUFFER = percentage | The percentage of free TCP send buffer space. |
| FREE TCP RECEIVE BUFFER = percentage | The percentage of free TCP receive buffer space. |
| FREE TCP OUT OF SEQUENCE BUFFER = percentage | The percentage of free TCP out of sequence buffer space. |

# Examples

The following sample output from the show ipv6 tcp connections command displays general information about each TCP connection on the router.

```
device# show ipv6 tcp connections
Local IP address:port <-> Remote IP address:port TCP state
10.168.182.110:23     <-> 10.168.8.186:4933 ESTABLISHED
10.168.182.110:8218   <-> 10.168.182.106:179 ESTABLISHED
10.168.182.110:8039   <-> 10.168.2.119:179 SYN-SENT
10.168.182.110:8159   <-> 10.168.2.102:179 SYN-SENT
2000:4::110:179       <-> 2000:4::106:8222 ESTABLISHED (1440)
Total 5 TCP connections
TCP MEMORY USAGE PERCENTAGE
FREE TCP = 98 percent
FREE TCP QUEUE BUFFER = 99 percent
FREE TCP SEND BUFFER = 97 percent
FREE TCP RECEIVE BUFFER = 100 percent
FREE TCP OUT OF SEQUENCE BUFFER = 100 percent
```

# show ipv6 tcp status

Displays detailed information about a specified TCP connection.

## Syntax

**shpw ipv6 tcp status** *local-ipv6-address local-port-num remote-ipv6-address remote-port-num*

## Parameters

*local-ipv6-address*
Specifies the IPv6 address of the local interface over which theTCP connection is taking place

*local-port-num*
Specifies the local port number over which a TCP connection is taking place.

*remote-ipv6-address*
Specifies the IPv6 address of the remote interface over which the TCP connection is taking place.

*remote-port-num*
Specifies the remote port number over which a TCP connection is taking place.

## Modes

User EXEC mode

## Command Output

The **show ipv6 tcp status** command displays the following information:

| Output field | Description |
|---|---|
| TCP = location | The location of the TCP. |
| Send: initial sequence number | The initial sequence number sent by the local router. |
| Send: first unacknowledged sequence number | The first unacknowledged sequence number sent by the local router. |
| Send: current send pointer | The current send pointer. |
| Send: next sequence number to send | The next sequence number sent by the local router. |
| Send: remote received window | The size of the remote received window. |
| Send: total unacknowledged sequence number | The total number of unacknowledged sequence numbers sent by the local router. |
| Send: total used buffers number | The total number of buffers used by the local router in setting up the TCP connection |
| Receive: initial incoming sequence number | The initial incoming sequence number received by the local router. |
| Receive: expected incoming sequence number | The incoming sequence number expected by the local router. |
| Receive: received window | The size of the local router's receive window. |
| Receive: bytes in receive queue | The number of bytes in the local router's receive queue. |
| Receive: congestion window | The size of the local router's receive congestion window. |

# Examples

The following sample output displays detailed information about TCP connection.

```
device# show ipv6 tcp status 2000:4::110 179 2000:4::106 8222
TCP: TCP = 0x217fc300
TCP: 2000:4::110:179 <-> 2000:4::106:8222: state: ESTABLISHED Port: 1
 Send: initial sequence number = 242365900
 Send: first unacknowledged sequence number = 242434080
 Send: current send pointer = 242434080
 Send: next sequence number to send = 242434080
 Send: remote received window = 16384
 Send: total unacknowledged sequence number = 0
 Send: total used buffers 0
 Receive: initial incoming sequence number = 740437769
    Receive: expected incoming sequence number = 740507227
    Receive: received window = 16384
    Receive: bytes in receive queue = 0
    Receive: congestion window = 1459
```

# show ipv6 traffic

Displays IPv6 traffic statistics.

## Syntax

**show ipv6 traffic**

## Modes

User EXEC mode

## Command Output

The **show ipv6 traffic** command displays the following information:

| Output Field | Description |
|---|---|
| IPv6 statistics | |
| received | The total number of IPv6 packets received by the router. |
| sent | The total number of IPv6 packets originated and sent by the router. |
| forwarded | The total number of IPv6 packets received by the router and forwarded to other routers. |
| delivered | The total number of IPv6 packets delivered to the upper layer protocol. |
| rawout | This information is used by Ruckus Technical Support. |
| bad vers | The number of IPv6 packets dropped by the router because the version number is not 6. |
| bad scope | The number of IPv6 packets dropped by the router because of a bad address scope. |
| bad options | The number of IPv6 packets dropped by the router because of bad options. |
| too many hdr | The number of IPv6 packets dropped by the router because the packets had too many headers. |
| no route | The number of IPv6 packets dropped by the router because there was no route. |
| can not forward | The number of IPv6 packets the router could not forward to another router. |
| redirect sent | This information is used by Ruckus Technical Support. |
| frag recv | The number of fragments received by the router. |
| frag dropped | The number of fragments dropped by the router. |
| frag timeout | The number of fragment timeouts that occurred. |
| frag overflow | The number of fragment overflows that occurred. |
| reassembled | The number of fragmented IPv6 packets that the router reassembled. |
| fragmented | The number of IPv6 packets fragmented by the router to accommodate the MTU of this router or of another device. |

| Output Field | Description |
|---|---|
| ofragments | The number of output fragments generated by the router. |
| can not frag | The number of IPv6 packets the router could not fragment. |
| too short | The number of IPv6 packets dropped because they are too short. |
| too small | The number of IPv6 packets dropped because they do not have enough data. |
| not member | The number of IPv6 packets dropped because the recipient is not a member of a multicast group. |
| no buffer | The number of IPv6 packets dropped because there is no buffer available. |
| forward cache miss | The number of IPv6 packets received for which there is no corresponding cache entry. |
| **ICMP6 statistics**<br>Some ICMP statistics apply to both Received and Sent, some apply to Received only, some apply to Sent only, and some apply to Sent Errors only. | |
| **Applies to received and sent** | |
| dest unreach | The number of Destination Unreachable messages sent or received by the router. |
| pkt too big | The number of Packet Too Big messages sent or received by the router. |
| time exceeded | The number of Time Exceeded messages sent or received by the router. |
| param prob | The number of Parameter Problem messages sent or received by the router. |
| echo req | The number of Echo Request messages sent or received by the router. |
| echo reply | The number of Echo Reply messages sent or received by the router. |
| mem query | The number of Group Membership Query messages sent or received by the router. |
| mem report | The number of Membership Report messages sent or received by the router. |
| mem red | The number of Membership Reduction messages sent or received by the router. |
| router soli | The number of Router Solicitation messages sent or received by the router. |
| router adv | The number of Router Advertisement messages sent or received by the router. |
| nei soli | The number of Neighbor Solicitation messages sent or received by the router. |
| nei adv | The number of Router Advertisement messages sent or received by the router. |
| redirect | The number of redirect messages sent or received by the router. |
| **Applies to received only** | |
| bad code | The number of Bad Code messages received by the router. |
| too short | The number of Too Short messages received by the router. |
| bad checksum | The number of Bad Checksum messages received by the router. |
| bad len | The number of Bad Length messages received by the router. |

| Output Field | Description |
|---|---|
| nd toomany opt | The number of Neighbor Discovery Too Many Options messages received by the router. |
| badhopcount | The number of Bad Hop Count messages received by the router. |
| **Applies to sent only** | |
| error | The number of Error messages sent by the router. |
| can not send error | The number of times the node encountered errors in ICMP error messages. |
| too freq | The number of times the node has exceeded the frequency of sending error messages. |
| **Applies to sent errors only** | |
| unreach no route | The number of Unreachable No Route errors sent by the router. |
| admin | The number of Admin errors sent by the router. |
| beyond scope | The number of Beyond Scope errors sent by the router. |
| address | The number of Address errors sent by the router. |
| no port | The number of No Port errors sent by the router. |
| pkt too big | The number of Packet Too Big errors sent by the router. |
| time exceed transit | The number of Time Exceed Transit errors sent by the router. |
| time exceed reassembly | The number of Time Exceed Reassembly errors sent by the router. |
| param problem header | The number of Parameter Problem Header errors sent by the router. |
| nextheader | The number of Next Header errors sent by the router. |
| option | The number of Option errors sent by the router. |
| redirect | The number of Redirect errors sent by the router. |
| unknown | The number of Unknown errors sent by the router. |
| **UDP statistics** | |
| received | The number of UDP packets received by the router. |
| sent | The number of UDP packets sent by the router. |
| no port | The number of UDP packets dropped because the packet did not contain a valid UDP port number. |
| input errors | This information is used by Ruckus Technical Support. |
| **TCP statistics** | |
| active opens | The number of TCP connections opened by the router by sending a TCP SYN to another device. |
| passive opens | The number of TCP connections opened by the router in response to connection requests (TCP SYNs) received from other devices. |
| failed attempts | This information is used by Ruckus Technical Support. |
| active resets | The number of TCP connections the router reset by sending a TCP RESET message to the device at the other end of the connection. |
| passive resets | The number of TCP connections the router reset because the device at the other end of the connection sent a TCP RESET message. |
| input errors | This information is used by Ruckus Technical Support. |
| in segments | The number of TCP segments received by the router. |

| Output Field | Description |
|---|---|
| out segments | The number of TCP segments sent by the router. |
| retransmission | The number of segments that the router retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment. |

# Examples

The following sample output displays the IPv6 traffic statistics.

```
device# show ipv6 traffic
IP6 Statistics
 36947 received, 66818 sent, 0 forwarded, 36867 delivered, 0 rawout
 0 bad vers, 23 bad scope, 0 bad options, 0 too many hdr
 0 no route, 0 can not forward, 0 redirect sent
 0 frag recv, 0 frag dropped, 0 frag timeout, 0 frag overflow
 0 reassembled, 0 fragmented, 0 ofragments, 0 can not frag
 0 too short, 0 too small, 11 not member
 0 no buffer, 66819 allocated, 21769 freed
 0 forward cache hit, 46 forward cache miss
ICMP6 Statistics
Received:
 0 dest unreach, 0 pkt too big, 0 time exceeded, 0 param prob
 2 echo req, 1 echo reply, 0 mem query, 0 mem report, 0 mem red
 0 router soli, 2393 router adv, 106 nei soli, 3700 nei adv, 0 redirect
 0 bad code, 0 too short, 0 bad checksum, 0 bad len
 0 reflect, 0 nd toomany opt, 0 badhopcount
Sent:
 0 dest unreach, 0 pkt too big, 0 time exceeded, 0 param prob
 1 echo req, 2 echo reply, 0 mem query, 0 mem report, 0 mem red
 0 router soli, 2423 router adv, 3754 nei soli, 102 nei adv, 0 redirect
 0 error, 0 can not send error, 0 too freq
Sent Errors:
 0 unreach no route, 0 admin, 0 beyond scope, 0 address, 0 no port
 0 pkt too big, 0 time exceed transit, 0 time exceed reassembly
 0 param problem header, 0 nextheader, 0 option, 0 redirect, 0 unknown
UDP Statistics
 470 received, 7851 sent, 6 no port, 0 input errors
TCP Statistics
 57913 active opens, 0 passive opens, 57882 failed attempts
 159 active resets, 0 passive resets, 0 input errors
 565189 in segments, 618152 out segments, 171337 retransmission
```

# show ipv6 tunnel

Displays a summary of IPv6 tunnel information.

## Syntax

**show ipv6 tunnel** [ **config** ]

## Parameters

**config**

Displays IPv6 tunnel configurations.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show ipv6 tunnel** command displays the following information.

| Output field | Description |
|---|---|
| Tunnel | The tunnel interface number. |
| Mode | The tunnel mode:<br>• configured: Indicates a manually configured tunnel. |
| Tunnel Status | The status of the tunnel.<br>• Active: Indicates that tunnel is in active state. |
| Packet Received | The number of packets received by a tunnel interface. Note that this is the number of packets received by the CPU. It does not include the number of packets processed in hardware. |
| Packet Sent | The number of packets sent by a tunnel interface. Note that this is the number of packets sent by the CPU. It does not include the number of packets processed in hardware. |

## Examples

The following is sample output from the **show ipv6 tunnel** command.

```
device# show ipv6 tunnel

IP6 Tunnels
Tunnel        Mode        Tunnel Status Packet Received    Packet Sent
1               configured     Active           0                              0
2               configured     Active           0                              22419
```

# show ipv6 tunnel traffic

Displays statistics about IPv6 tunnel traffic.

## Syntax

**show ipv6 tunnel traffic**

## Modes

User EXEC mode

## Examples

The following is sample output from the **show ipv6 tunnel traffic** command.

```
device> show ipv6 tunnel traffic

IPSEC Tunnels
 Tunnel Status Packet Received Packet Sent Bytes Received Bytes Sent
 1 up/up 85530501 42778752 360787126058 180378526620
 9 up/up 37984 45673 8079092 9180122
 18 up/up 29804 29530 6688012 6435816
…
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show ipv6 vrrp

Displays information about IPv6 Virtual Router Redundancy Protocol (VRRP) sessions.

## Syntax

**show ipv6 vrrp** [ **brief** ]

**show ipv6 vrrp** [ **ethernet** *unit***/***slot***/***port* | **ve** *num* ]

**show ipv6 vrrp** [ **statistics** [ **ethernet** *unit***/***slot***/***port* | **ve** *num* ] ]

**show ipv6 vrrp** [ **ve** *num* [ **vrid** *VRID* ] ]

**show ipv6 vrrp** [ **vrid** *VRID* [ **ethernet** *unit* **/** *slot* **/** *port* | **ve** *num* ] ]

## Parameters

**brief**

Displays summary information about the IPv6 VRRP session.

**ethernet** *unit slot port*

Displays IPv6 VRRP information only for the specified Ethernet port. A forward slash "/" must be entered between the *unit*, *slot*, and *port* variables.

**ve** *num*

Displays IPv6 VRRP information only for the specified virtual Ethernet port.

**statistics**

Displays statistical information about the IPv6 VRRP session.

**vrid** *VRID*

Displays IPv6 VRRP information only for the specified virtual router ID (VRID).

## Modes

User EXEC mode

## Usage Guidelines

This command can be entered in any mode. This command supports IPv6 VRRP; to display information about VRRP Extended (VRRP-E) sessions, use the **show ipv6 vrrp-extended** command.

## Command Output

The following is a partial list of output field descriptions for the **show ipv6 vrrp** command.

| Output field | Description |
|---|---|
| Total number of VRRP routers defined | The total number of virtual routers configured and currently running on this Ruckus ICX device. For example, if the device is running VRRP-E, the total applies only to VRRP-E routers. |

| Output field | Description |
|---|---|
| Interface | The interface on which VRRP is configured. If VRRP is configured on multiple interfaces, information for each interface is listed separately. |
| VRID | The ID of the virtual router configured on this interface. If multiple virtual routers are configured on the interface, information for each virtual router is listed in a separate row. |
| state | This device's VRRP state for the virtual router. The state can be one of the following:<br>• init—The virtual router is not enabled (activated). If the state remains init after you activate the virtual router, make sure that the virtual router is also configured on the other routers and that the routers can communicate with each other.<br>If the state is init and the mode is incomplete, make sure you have specified the IP address for the virtual router.<br>• backup—This device is a backup for the virtual router.<br>• master—This device is the master for the virtual router. |
| current priority | The current VRRP priority of this device for the virtual router. |
| preempt-mode | Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains a "true." If the mode is disabled, this field is blank. |

# Examples

The following example displays IPv6 VRRP session information in detail.

```
device(config)# show ipv6 vrrp

Total number of VRRP routers defined: 1
Interface 1/1/3
----------------
auth-type no authentication
VRID 13 (index 2)
interface 1/1/3
state master
administrative-status enabled
version v3
mode non-owner(backup)
virtual mac 0000.5e00.0217
priority 100
current priority 100
track-priority 1
hello-interval 1000 ms
backup hello-interval 60000 ms
advertise backup disabled
dead-interval 3000 ms
preempt-mode true
ipv6-address 3013::1
next hello sent in 700 ms
short-path-forwarding disabled
```

The following example displays IPv6 VRRP statistical information.

```
device# show ipv6 vrrp statistics

Global IPv6 VRRP statistics
-------------------------------
- received vrrp packets with checksum errors = 0
- received vrrp packets with invalid version number = 0
- received vrrp packets with unknown or inactive vrid = 0
Interface 1/1/3
----------------
VRID 13
- number of transitions to backup state = 1
- number of transitions to master state = 1
- total number of vrrp packets received = 0
. received backup advertisements = 19
. received packets with zero priority = 0
. received packets with invalid type = 0
. received packets with invalid authentication type = 0
. received packets with authentication type mismatch = 0
. received packets with authentication failures = 0
. received packets dropped by owner = 0
. received packets with ttl errors = 0
. received packets with ipv6 address mismatch = 0
. received packets with advertisement interval mismatch = 0
. received packets with invalid length = 0
- total number of vrrp packets sent = 1175
. sent backup advertisements = 0
. sent packets with zero priority = 0
- received neighbor solicitation packets dropped = 0
- received proxy neighbor solicitation packets dropped = 0
- received ipv6 packets dropped = 0
```

The following example displays IPv6 VRRP configuration information about VRID 1.

```
device# show ipv6 vrrp vrid 1

Interface 1/1/1
----------------
auth-type no authentication
VRID 1 (index 1)
interface 1/1/1
state master
administrative-status enabled
version v3
mode non-owner(backup)
virtual mac dddd.eeee.ffff (configured)
priority 100
current priority 100
track-priority 1
hello-interval 1000 ms
backup hello-interval 60000 ms
advertise backup disabled
dead-interval 3600 ms
preempt-mode true
ipv6 address 10:20:1::100
next hello sent in 400 ms
```

The following example displays an auto-generated IPv6 virtual link-local address used in the VRRPv3 VRID 1 instance.

**NOTE**
This example is applicable only to the auto-generation of an IPv6 virtual link-local address.

```
device# show ipv6 vrrp vrid 1

VRID 1 (index 1)
 interface 1/1/1
 state master
 administrative-status enabled
 version v3
 mode owner
 virtual mac 0000.5e00.0101
 virtual link-local fe80::200:5eff:fe00:201
 priority 255
 current priority 255
 track-priority 2
 hello-interval 1000 ms
 backup hello-interval 60000 ms
 number of configured virtual address 2
 ipv6-address 1:2:45::2
 ipv6-address 1:2:46::2
 next hello sent in 300 ms
 Track MCT-VPLS-State: Disable
```

# show ipv6 vrrp-extended

Displays information about IPv6 Virtual Router Redundancy Protocol Extended (VRRP-E) sessions.

## Syntax

**show ipv6 vrrp-extended** [ **brief** ]

**show ipv6 vrrp-extended** [ **ethernet** *unit**/**slot**/**port* | **ve** *num* ]

**show ipv6 vrrp-extended** [ **statistics** [ **ethernet** *unit**/**slot**/**port* | **ve** *num* ] ]

**show ipv6 vrrp-extended** [ **ve** *num* [ **vrid** *VRID* ] ]

**show ipv6 vrrp-extended** [ **vrid** *VRID* [ **ethernet** *unit**/**slot**/**port* | **ve** *num* ] ]

## Parameters

**brief**

Displays summary information about the IPv6 VRRP-E session.

**ethernet** *unit slot port*

Displays IPv6 VRRP information only for the specified Ethernet port. A forward slash "/" must be entered between the *unit*, *slot*, and *port* variables.

**statistics**

Displays statistical information about the IPv6 VRRP-E session.

**ve** *num*

Displays IPv6 VRRP-E information only for the specified virtual Ethernet port.

**vrid** *VRID*

Displays IPv4 VRRP-E information only for the specified virtual-group ID.

## Modes

User EXEC mode

## Usage Guidelines

Use this command to display information about IPv6 VRRP-E sessions, either in summary or full-detail format. You can also specify a virtual group or interface for which to display output.

This command supports IPv6 VRRP-E. You can modify or redirect the displayed information by using the default Linux tokens (|, >).

## Command Output

The **show ipv6 vrrp-extended** command displays the following information:

| Output field | Description |
|---|---|
| Total number of VRRP-E routers defined | The total number of virtual routers configured on this Ruckus ICX device. |

| Output field | Description |
|---|---|
| | **NOTE**<br>The total applies only to the protocol the device is running. For example, if the device is running VRRP-E, the total applies only to VRRP-E routers. |
| Interface | The interface on which VRRP-E is configured. If VRRP-E is configured on multiple interfaces, information for each interface is listed separately. |
| VRID | The ID of the virtual router configured on this interface. If multiple virtual routers are configured on the interface, information for each virtual router is listed in a separate row. |
| Current Priority | The current VRRP-E priority of this device for the virtual router. |
| Flags | Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains a "P". If the mode is disabled, this field is blank.<br>• P:Preempt 2:V2 3:V3<br>• 2: implies VRRP Version2<br>• 3: implies VRRP Version3 |
| Short-Path-Fwd | This Ruckus device's VRRP state for the virtual router. The state can be one of the following:<br>• Init—The virtual router is not enabled (activated). If the state remains Init after you activate the virtual router, make sure that the virtual router is also configured on the other routers and that the routers can communicate with each other.<br>**NOTE**<br>If the state is Init and the mode is incomplete, make sure you have specified the IP address for the virtual router.<br>• Backup—This device is a backup for the virtual router.<br>• Master—This device is the master for the virtual router. |
| Master IP Address | The IPv6 address of the router interface that is currently the Master for the virtual router. |
| Backup IP Address | The IPv6 addresses of the router interfaces that are currently backups for the virtual router. |
| Virtual IP Address | The virtual IPv6 address that is being backed up by the virtual router. |

# Examples

The following example displays summary information for an IPv6 VRRP-E session.

```
device(config)# show ipv6 vrrp-extended brief

Total number of VRRP routers defined: 1
Flags Codes - P:Preempt 2:V2 3:V3 S:Short-Path-Fwd
Intf   VRID  CurrPrio Flags State   Master-IPv6 Backup-IPv6 Virtual-IPv6
                                    Address     Address     Address
-------------------------------------------------------------------------
1/1/3  2     100      P3-  Master  Local       3013::2     3013::99
```

The following example displays detailed IPv6 VRRP-E configuration information about VRID 1.

```
device# show ipv6 vrrp-extended vrid 1

Interface 1/1/1
---------------
auth-type md5-authentication
VRID 1 (index 1)
interface 1/1/1
state master
administrative-status enabled
mode non-owner(backup)
virtual mac dddd.eeee.ffff (configured)
priority 100
current priority 100
track-priority 5
hello-interval 1 sec
backup hello-interval 60 sec
advertise backup disabled
dead-interval 0 ms
preempt-mode true
virtual ipv6 address 10:20:1::100
```

# show issu errors

Displays stack upgrade error information when an upgrade is in progress.

## Syntax

**show issu errors**

## Modes

Privileged EXEC mode

## Examples

Use the following command to get ISSU error information.

```
device# show issu errors
ISSU State: UPGRADE ABORT
Abort reason: UNABLE TO UPGRADE UNIT
Unit 1 did not join the stack after upgrade
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support for Campus Fabric (SPX) systems was added. |

# show issu sequence

Displays the sequence in which units will be upgraded.

## Syntax

**show issu sequence**

## Modes

Privileged EXEC mode

## Command Output

The **show issu sequence** command displays the following information:

| Output field | Description |
|---|---|
| ID | The stack unit. |
| Type | Platform and model. |
| Role | active, member, or standby |

## Examples

Use this command to display the sequence of the stack unit upgrade.

```
device# show issu sequence
Stack units will be upgraded in the following order
ID      Type              Role
1       ICX7450-32ZP      standby
3       ICX7450-32ZP      member
4       ICX7450-32ZP      active
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support for Campus Fabric (SPX) systems was added. |

# show issu status

Runs a pre-ISSU check and monitors the status of the current upgrade.

## Syntax

**show issu status**

## Modes

Privileged EXEC mode

## Usage Guidelines

Use this command to show the ISSU status before or during an upgrade.

## Examples

Output for a successful upgrade in progress.

```
device# show issu status
ISSU Status: In Progress
Upgrade State: UNIT JOIN

Upgrade Option: issu primary
ID   Type          Role      State
1    ICX7450-32ZP  member    UPGRADING
3    ICX7450-32ZP  member    UPGRADE PENDING
4    ICX7450-32ZP  active    UPGRADE PENDING
```

Output of the command when errors are encountered.

```
device# show issu status
ISSU Status: Aborted
Upgrade State: UPGRADE ABORT
Upgrade Option: issu primary
Reason for Abort: UNABLE TO UPGRADE

ID   Type          Role      State
1    ICX7450-32ZP  member    UPGRADE ABORT
3    ICX7450-32ZP  standby   UPGRADE PENDING
4    ICX7450-32ZP  active    UPGRADE PENDING
```

**NOTE**

An error condition is indicated by three asterisks (***).

If a manual abort is done or ISSU detects an abort condition (with ISSU started with the no **on-error** option), the stack is left as it is and a manual recovery is required by running either the **reload-primary** or **reload-secondary** command.

If an upgrade is not in progress, this command displays information about whether the system is ready for an upgrade.

```
device# show issu status
Topology is Ring                          Yes
Standby Present                           Yes
Standby ready for upgrade                 Yes
Flash use in progress                     No
Secure Setup in progress                  No
ISSU in progress or aborted               No
Election pending                          No
Election in progress                      No
Reload pending                            No
CPU utilization high                      No
All units in ready state                  Yes
Primary Image is upgrade compatible       Yes
Secondary Image is upgrade compatible     Yes
Startup config and Running Config Same    Yes
User in Config mode                       No
System ready for issu
ISSU not in progress
```

If an upgrade is completed, this command displays the following information,

```
device# show issu status
Last upgrade time                         00:02:19.367 GMT+00 Tue Mar 20 2016
The older image before-ISSU               SPR08050b433.bin
Topology is Ring                          Yes
Standby Present                           Yes
Standby ready for upgrade                 Yes
Flash use in progress                     No
Secure Setup in progress                  No
ISSU in progress or aborted               No
Election pending                          No
Election in progress                      No
Reload pending                            No
CPU utilization high                      No
All units in ready state                  Yes
Primary Image is upgrade compatible       Yes
Secondary Image is upgrade compatible     Yes
Startup config and Running Config Same    Yes
User in Config mode                       No
System ready for issu
ISSU not in progress
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | Support for Campus Fabric (SPX) systems was added. |

# show keychain

Displays keychain-related configuration and status information.

## Syntax

**show keychain** [ **resource** | **name** *keychain-name* [ *key-id* | **active** ] ]

## Parameters

**resource**

Displays the number of keychains configured, the status of the keychain timer, and the number of keys configured currently.

**name** *keychain-name*

Displays the keychain configuration details of a specific keychain.

*key-id*

Displays the details of a specific key within a keychain.

**active**

Displays the active keys under a specific keychain.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Keychain configuration mode

Key ID configuration mode

## Examples

The following example displays keychain configuration details.

```
device# show keychain
Keychain: ruckus
Tolerance: 8639999
-------------------------------------------------------------------------
  Key-id  |    Algo    | SendActive | SendTimer| AcceptActive | AcceptTimer
-------------------------------------------------------------------------
      1      hmac-sha-256  Yes(GMT+00)    8647414    Yes(GMT+00)    8561014
      2      sha-256       No(GMT+00)          -     No(GMT+00)          -
      3      sha-256       No(GMT+00)          -     No(GMT+00)          -
      4      None          No(GMT+00)          -     No(GMT+00)
```

The following example displays the number of keychains configured, the status of the keychain timer, and the number of keys configured currently.

```
device# show keychain resource
Total Keychains Configured: 1
Keychain Timer Operational: Yes
Keychain Resource Infromation:-
                     alloc in-use  avail get-fail    limit  get-mem  size init
Keychain                64      1     63        0       64        1   339   64
Key                    256      2    254        0     1024        2   983  256
Misc                    64      0     64        0      128        0    16   64
```

The following example displays the keychain configuration details of a specific keychain.

```
device# show keychain name ruckus
Keychain: ruckus
  Key-id   : 1
      AuthAlgorithm: hmac-sha-256
      Key-String   : *******
      Send Lifetime:-
          Start    : 09-19-2017 10:55:00 End         : 09-19-2017 13:00:00
          Active   : Yes                 TimeToExpire: 8647208 sec
          Timezone : GMT+00
      Accept Lifetime:-
          Start    : 09-18-2017 12:00:00 End         : 09-18-2017 13:00:00
          Active   : Yes                 TimeToExpire: 8560808 sec
          Timezone : GMT+00
```

The following example displays the details of a specific key by specifying the key ID within a keychain.

```
device# show keychain name ruckus 1
Keychain: ruckus
  Key-id   : 1
      AuthAlgorithm: hmac-sha-256
      Key-String   : *******
      Send Lifetime:-
          Start    : 09-19-2017 10:55:00 End         : 09-19-2017 13:00:00
          Active   : Yes                 TimeToExpire: 8647018 sec
          Timezone : GMT+00
      Accept Lifetime:-
          Start    : 09-18-2017 12:00:00 End         : 09-18-2017 13:00:00
          Active   : Yes                 TimeToExpire: 8560618 sec
          Timezone : GMT+00
```

The following example displays the active keys under the "ruckus" keychain.

```
device# show keychain name ruckus active
Keychain: ruckus
  Key-id   : 1
      AuthAlgorithm: hmac-sha-256
      Key-String   : *******
      Send Lifetime:-
          Start    : 09-19-2017 10:55:00 End         : 09-19-2017 13:00:00
          Active   : Yes                 TimeToExpire: 8646948 sec
          Timezone : GMT+00
      Accept Lifetime:-
          Start    : 09-18-2017 12:00:00 End         : 09-18-2017 13:00:00
          Active   : Yes                 TimeToExpire: 8560548 sec
          Timezone : GMT+00
```

# History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# show lag

Displays Link Aggregation Group (LAG) information.

## Syntax

**show lag** [ *lag-name* | **brief** | **deployed** | **dynamic** | **id** *number* | **keep-alive** | **static** ]

## Parameters

*lag-name*
Displays the LAG specified by the LAG name.

**brief**
Displays the LAG information summary.

**deployed**
Displays information about all the deployed LAGs.

**dynamic**
Displays information about dynamic LAGs.

**id** *number*
Displays information about the LAG specified by the ID number.

**keep-alive**
Displays information about keep-alive LAGs.

**static**
Displays information about static LAGs.

## Modes

User EXEC mode

Privileged EXEC mode

LAG configuration mode

## Command Output

The **show lag** command displays the following information:

| Output field | Description |
|---|---|
| Total number of LAGS | The total number of LAGs that have been configured on the device. |
| Total number of deployed LAGS | The total number of LAGs on the device that are currently deployed. |
| Total number of trunks created | The total number of trunks that have been created on the LAG. The total number of LAGs available are shown also. Because keep-alive LAGs do not use LAG IDs, they are not listed and do not subtract from the number of LAGs available. |
| LACP System Priority /ID | The system priority configured for the device. The ID is the system priority that is the base MAC address of the device. |

| Output field | Description |
|---|---|
| LACP Long timeout | The number of seconds used for the LACP long timeout mode. This is only applicable for dynamic or keep-alive LAGs. |
| LACP Short timeout | The number of seconds used for the LACP short timeout mode. This is only applicable for dynamic or keep-alive LAGs. |

The following information is displayed per-LAG in the **show lag brief** command:

| Output field | Description |
|---|---|
| LAG | The name of the LAG, LAG ID number, the configured type of the LAG: static, dynamic, or keep-alive, status of LAG deployment: deployed or not. |

The following information is displayed per-LAG in the **show lag** command for each LAG configured:

| Output field | Description |
|---|---|
| **LAG Configuration** | |
| Ports | List of ports configured with the LAG. |
| Port Count | Number of ports configured on the LAG. |
| Lag Interface | The LAG virtual interface. |
| Trunk Type | The load sharing method configured for the LAG. The trunk types are hash-based and resilient-hash. |
| LACP Key | The link aggregation key for the LAG. |

The following information is displayed for the **show lag deployed** command:

| Output field | Description |
|---|---|
| **Deployment** | |
| LAG ID | The LAG ID number. |
| Active Primary | The port within the LAG where most protocol packets are transmitted. This is not the same as the configured primary port of the LAG. |
| Port | The chassis slot and port number of the interface. |
| Link | The status of the link, which can be one of the following:<br>• up<br>• down |
| State | The Layer 2 state for the port. |
| Dupl | The duplex state of the port, which can be one of the following:<br>• Full<br>• Half<br>• None |
| Speed | The bandwidth of the interface. |
| Trunk | The LAG ID of the port. |
| Tag | Indicates whether the ports have 802.1q VLAN tagging. The value can be Yes or No. |
| Pri | Indicates the Quality of Service (QoS) priority of the ports. The priority can be a value from 0 through 7. |
| MAC | The MAC address of the port. |

| Output field | Description |
|---|---|
| Name | The name (if any) configured for the port. |
| Sys P | Lists the system priority configured for the device. |
| Port P | Lists the port's link aggregation priority. |
| Key | Lists the link aggregation key. |
| Act | Indicates the link aggregation mode, which can be one of the following:<br>• No: The mode is passive on the port. If link aggregation is enabled (and the mode is passive), the port can send and receive LACPDU messages to participate in negotiation of an aggregate link initiated by another port, but cannot search for a link aggregation port or initiate negotiation of an aggregate link.<br>• Yes: The mode is active. The port can send and receive LACPDU messages. |
| Tio | Indicates the timeout value of the port. The timeout value can be one of the following:<br>• L: Long. The LAG group has already been formed and the port is therefore using a longer message timeout for the LACPDU messages exchanged with the remote port. Typically, these messages are used as confirmation of the health of the aggregate link.<br>• S: Short. The port has just started the LACPDU message exchange process with the port at the other end of the link. The S timeout value also can mean that the link aggregation information received from the remote port has expired and the ports are starting a new information exchange. |
| Agg | Indicates the link aggregation state of the port. The state can be one of the following:<br>• Agg: Link aggregation is enabled on the port.<br>• No: Link aggregation is disabled on the port. |
| Syn | Indicates the synchronization state of the port. The state can be one of the following:<br>• No: The port is out of sync with the remote port. The port does not understand the status of the LACPDU process and is not prepared to enter a LAG link.<br>• Syn: The port is in sync with the remote port. The port understands the status of the LACPDU message exchange process, and therefore knows the LAG group to which it belongs, the link aggregation state of the remote port, and so on. |
| Dis | Indicates the collection state of the port, which determines whether the port is ready to send traffic over the LAG link:<br>• Col: The port is ready to send traffic over the LAG link.<br>• No: The port is not ready to send traffic over the LAG link. |
| Col | Indicates the distribution state of the port, which determines whether the port is ready to receive traffic over the LAG link:<br>• Dis: The port is ready to receive traffic over the LAG link.<br>• No: The port is not ready to receive traffic over the LAG link. |
| Def | Indicates whether the port is using default link aggregation values. The port uses default values if it has not received link aggregation information through LACP from the port at the remote end of the link. This field can have one of the following values:<br>• Def: The port has not received link aggregation values from the port at the other end of the link and is therefore using its default link aggregation LACP settings.<br>• No: The port has received link aggregation information from the port at the other end of the link and is using the settings negotiated with that port. |
| Exp | Indicates whether the negotiated link aggregation settings have expired. The settings expire if the port does not receive an LACPDU message from the port at the other end of the link before the message timer expires. This field can have one of the following values:<br>• Exp: The link aggregation settings this port negotiated with the port at the other end of the link have expired. The port is now using its default link aggregation settings. |

| Output field | Description |
|---|---|
| | • No: The link aggregation values that this port negotiated with the port at the other end of the link have not expired. The port is still using the negotiated settings. |
| Ope | • Ope (operational): The port is operating normally.<br>• Blo (blocked): The port is blocked because the adjacent port is not configured with link aggregation or because it is not able to join a LAG. An LACP port is blocked until it becomes part of a LAG. Also, an LACP port is blocked if its state becomes "default". To unblock the port and bring it to an operational state, enable link aggregation on the adjacent port and ensure that the ports have the same key.<br>• Frc (force-up): The port is in "force-up" mode. If you have configured the **force-up ethernet** command on the member port of a dynamic LAG, the port goes into "force-up" mode and is logically operational when the dynamic LAG is not operating.<br>• Err: If there is a peer information mismatch, then that particular port is moved to the Error disable state (Err). |
| Port | The chassis slot and port number of the interface. |
| Partner System ID | The partner system ID indicating the system's priority and the MAC address of the port. |
| Partner Key | The partner key value. Valid key values range from 1 to 65535. |
| LACP Rx Count | This is the counter for LACPDUs received on this port. |
| LACP Tx Count | This is the counter for LACPDUs transmitted from this port. |

# Examples

The following example shows sample output of the **show lag** command.

```
device# show lag
Total number of LAGs:          2
Total number of deployed LAGs: 2
Total number of trunks created:2 (126 available)
LACP System Priority / ID:     1 / 609c.9fbc.bf14
LACP Long timeout:             120, default: 120
LACP Short timeout:            3, default: 3
=== LAG "tosp12" ID 1 (dynamic Deployed) ===
LAG Configuration:
   Ports:         e 1/1/5 e 1/1/7
   Port Count:    2
   Lag Interface: lg1
   Trunk Type:    hash-based
   LACP Key:      20001
Deployment: HW Trunk ID 1
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/5      Down    None    None None  1     No  1    0   609c.9fbc.bf14

1/1/7      Disable None    None None  1     No  1    0   609c.9fbc.bf14

Port       [Sys P] [Port P] [ Key ] [Act][Tio][Agg][Syn][Col][Dis][Def][Exp][Ope]
1/1/5           1        1   20001   Yes   S   Agg  Syn  No   No   Def  No   Dwn
1/1/7           1        1   20001   Yes   S   Agg  Syn  No   No   Def  No   Dwn
 Partner Info and PDU Statistics
Port          Partner         Partner     LACP      LACP
            System ID          Key     Rx Count  Tx Count
1/1/5    1-0000.0000.0000        4        0         0
1/1/7    1-0000.0000.0000        6        0         0
=== LAG "tosp16" ID 2 (static Deployed) ===
LAG Configuration:
   Ports:         e 1/1/6 e 1/1/8
   Port Count:    2
   Lag Interface: lg2
   Trunk Type:    hash-based
Deployment: HW Trunk ID 2
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/6      Down    None    None None  2     No  1    0   609c.9fbc.bf14

1/1/8      Down    None    None None  2     No  1    0   609c.9fbc.bf14
```

The following example shows sample output of the **show lag deployed** command.

```
device(config)# show lag tosp16
Total number of LAGs:          4
Total number of deployed LAGs: 2
Total number of trunks created:2 (126 available)
LACP System Priority / ID:     1 / 609c.9fbc.bf14
LACP Long timeout:             120, default: 120
LACP Short timeout:            3, default: 3
=== LAG "tosp16" ID 2 (static Deployed) ===
LAG Configuration:
   Ports:         e 1/1/6 e 1/1/8
   Port Count:    2
   Lag Interface: lg2
   Trunk Type:    hash-based
Deployment: HW Trunk ID 2
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/6      Down    None    None None  2     No  1    0   609c.9fbc.bf14

1/1/8      Down    None    None None  2     No  1    0   609c.9fbc.bf14
```

The following example shows sample output of the **show lag** command with the "resilient-hash" trunk type in the LAG configuration.

```
device(config)# show lag id 1
Total number of LAGs:          4
Total number of deployed LAGs: 2
Total number of trunks created:2 (126 available)
LACP System Priority / ID:     1 / 609c.9fbc.bf14
LACP Long timeout:             120, default: 120
LACP Short timeout:            3, default: 3
=== LAG "tosp16" ID 2 (static Deployed) ===
LAG Configuration:
   Ports:         e 1/1/6 e 1/1/8
   Port Count:    2
   Lag Interface: lg2
   Trunk Type:    hash-based
Deployment: HW Trunk ID 2
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/6      Down    None    None None  2     No  1    0   609c.9fbc.bf14

1/1/8      Down    None    None None  2     No  1    0   609c.9fbc.bf14
```

# History

| Release version | Command history |
|---|---|
| 08.0.30d | This command was modified to display a changed output for the **deployed** keyword. |
| 08.0.50 | This command was modified to display a changed output for the "resilient-hash" trunk type in the LAG configuration. |
| 08.0.61 | This command was modified to include LAG ID options. |

# show license

Displays information about Self Authenticated Upgrade licenses installed on a device.

## Syntax

**show license**

## Modes

Privileged EXEC mode

## Usage Guidelines

The command can be used on a standalone device or on the active controller for a stack.

This command does not display information about XML licenses that were installed on the device in FastIron 08.0.70 or earlier releases. The **show license unit** command displays information about both SAU and XML licenses.

## Command Output

The **show license** command displays the following information:

| Output field | Description |
|---|---|
| Unit | Unit number assigned in the stack. For standalone units, the unit number is 1. |
| License Name | Name of Software Authenticated Upgrade (SAU) license installed. |
| L3 Premium | (Yes, No) Indicates whether Layer 3 features are enabled by the license. |
| Port Speed Upgrade | (Yes, No) Indicates whether the license allows ports to be upgraded from the default speed (1 Gbps). |
| Speed | Speed to which ports covered by the license can be upgraded (10 Gbps). |
| Ports | Indicates the number of ports covered by the license. |
| MACsec | Indicates whether a MACsec license is installed. |

## Examples

The following example shows a 2x10G license installed on stack unit 1. Stack unit 2 has 8x10G and Layer 3 Premium licenses.

```
ICX7250-24P Router# show license installed
Unit  License Name      L3 Premium  Port Speed Upgrade Speed    Ports   MACsec
1      L3-PREM-2X10G    Yes         Yes                10G      2       NA
2      L3-PREM-8x10G    Yes         Yes                10G      8       NA
ICX7150-24P Router#
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was enhanced to support all ICX platforms that support SAU licensing. |

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show license installed

Displays detailed information about Self Authenticated Upgrade licenses installed on a device.

## Syntax

**show license installed**

## Modes

Privileged EXEC mode

## Usage Guidelines

The command can be used on a standalone device or on the active controller for a stack.

## Command Output

The **show license installed** command displays the following information:

| Output field | Description |
|---|---|
| Unit | Unit number assigned in the stack. For standalone units, the unit number is 1. |
| License Name | Name of Software Authenticated Upgrade (SAU) license installed. |
| L3 Premium | (Yes, No) Indicates whether Layer 3 features are enabled by the license. |
| Port Speed Upgrade | (Yes, No) Indicates whether the license allows ports to be upgraded from the default speed (1 Gbps). |
| Speed | Speed to which ports covered by the license can be upgraded (10 Gbps). |
| Ports | Indicates the number of ports covered by the license. |
| MACsec | Indicates whether a MACsec license is installed. |
| SerialNo (L3/ICX7150) | License serial number. ICX7150: the serial number for ICX7150 platform; L3: L3 license serial number for all other ICX platform |
| SerialNo(PoD/MACsec) | License serial number. PoD: PoD license serial number for ICX7250 platform; MACsec: MACsec license serial number for ICX7450/IX7650 platforms. |

## Examples

The following example shows a 2x10G license installed on stack unit 1. Stack unit 2 has 8x10G and Layer 3 Premium licenses.

```
ICX7250-24P Router# show license installed
Unit  License Name   L3 Premium  Port Speed Upgrade Speed  Ports  MACsec SerialNo (L3/ICX7150)
SerialNo(PoD/MACsec)
1     L3-PREM-2X10G  Yes         Yes                10G    2      NA     PR320400289
PR320400290
2     L3-PREM-8x10G  Yes         Yes                10G    8      NA     PR320400291
ICX7250-24P Router#
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was enhanced to support all ICX platforms that support SAU licensing. |
| 08.0.61 | This command was introduced. |

# show license node-locked

Displays information about all node-locked software licenses on a device.

## Syntax

**show license node-locked**

## Modes

Privileged EXEC level

## Usage Guidelines

This command can be used to display information about node-locked XML software licenses on a device that were installed on a device in FastIron 08.0.70 or earlier releases. Use the **show license installed** command to display information about SAU licenses.

## Command Output

The **show license node-locked** command displays the following information:

| Output field | Description |
|---|---|
| Index | The index number specifies the software license file for a specific stack. The index number is generated by the member unit. |
| Lid | The license ID. This number is embedded in the Ruckus device. |
| Lic Mode | Indicates whether the license is a non-node-locked license or node-locked license. |
| License name | The name of the license installed for the license index number on the stack unit. |
| Lid/Serial No | The license ID. The number is embedded in the Ruckus device. The serial number for only a non-node locked license. The serial number is generated when you request a license through the license portal. The serial number is not the device name. |
| License Type | Indicates whether the license is normal (permanent) or trial (temporary). |
| Status | Indicates the status of the license:<br>• Valid - A license is valid if the LID matches the license ID of the device for which the license was purchased, and the package name is recognized by the system.<br>• Invalid - The LID does not match the license ID of the device for which the license was purchased.<br>• Active - The license is valid and in effect on the device. |

| Output field | Description |
|---|---|
| | • Not used - The license is not in effect on the device.<br>• Expired - For trial licenses only, this indicates that the trial license has expired.<br>• Duplicated - For non-node-locked licenses, this indicates that the same serial number is used for devices in a stacking system. |
| License Period | If the license type is trial (temporary), this field displays the number of days the license is valid. If the license type is normal (permanent), this field displays Unlimited. |
| License Capacity | The port capacity of the Ports of Demand (PoD) license. |
| Trial license information | Indicates the trial license information details as displayed in the **show license** command output.<br>• days used - The number of days the trial license has been effect.<br>• hours used - The number of hours the trail license has been in effect.<br>• days left - The number of days left before the trial license expires.<br>• hours left - The number of hours left before the trial license expires. |

# Examples

The following **show license node-locked** command output displays software licensing information. The hardware license information is not displayed.

```
ICX7250-24 Router# show license node-locked
Index    Lic Mode        Lic Name              Lid           Lic Type    Status     Lic Period    Lic
Capacity
Stack unit 1:
2        Node Lock       ICX7250-10G-LIC-POD   fwjINHGnFMF   Normal      Active     Unlimited
2
Stack unit 2:
1        Node Lock       ICX7250-10G-LIC-POD   fwlINJKnFhx   Normal      Active     Unlimited
8
ICX7250-24 Router#
```

# show license non-node-locked

Displays information about all non-node-locked software licenses on a device.

## Syntax

**show license non-node-locked**

## Modes

Privileged EXEC level

## Usage Guidelines

This command can be used to display information about non-node-locked XML software licenses that were installed on a device in FastIron 08.0.70 or earlier releases. Use the **show license installed** command to display information about SAU licenses.

## Command Output

The **show license non-node-locked** command displays the following information:

| Output field | Description |
|---|---|
| Index | The index number specifies the software license file for a specific stack. The index number is generated by the member unit. |
| Lid | The license ID. This number is embedded in the Ruckus device. |
| Lic Mode | Indicates whether the license is a non-node-locked license or node-locked license. |
| License name | The name of the license installed for the license index number on the stack unit. |
| Lid/Serial No | The license ID. The number is embedded in the Ruckus device. The serial number for only a non-node locked license. The serial number is generated when you request a license through the license portal. The serial number is not the device name. |
| License Type | Indicates whether the license is normal (permanent) or trial (temporary). |
| Status | Indicates the status of the license:<br>• Valid - A license is valid if the LID matches the license ID of the device for which the license was purchased, and the package name is recognized by the system.<br>• Invalid - The LID does not match the license ID of the device for which the license was purchased.<br>• Active - The license is valid and in effect on the device. |

| Output field | Description |
|---|---|
| | • Not used - The license is not in effect on the device.<br>• Expired - For trial licenses only, this indicates that the trial license has expired.<br>• Duplicated - For non-node-locked licenses, this indicates that the same serial number is used for devices in a stacking system. |
| License Period | If the license type is trial (temporary), this field displays the number of days the license is valid. If the license type is normal (permanent), this field displays Unlimited. |
| License Capacity | The port capacity of the Ports of Demand (PoD) license. |
| Trial license information | Indicates the trial license information details as displayed in the **show license** command output.<br>• days used - The number of days the trial license has been effect.<br>• hours used - The number of hours the trail license has been in effect.<br>• days left - The number of days left before the trial license expires.<br>• hours left - The number of hours left before the trial license expires. |

# Examples

The following **show license non-node-locked** command output displays software licensing information. The hardware license information is not displayed.

```
ICX7250-24 Router# show license non-node-locked
Index    Lic Mode        Lic Name              Serial Number  Lic Type    Status     Lic Period    Lic
Capacity
Stack unit 1:
1        Non-Node Lock   ICX7250-PREM-LIC-SW    EN0E583FD98    Normal      Active     Unlimited
1
Stack unit 2:
2        Non-Node Lock   ICX7250-PREM-LIC-SW    EN0C606AA2E    Normal      Active     Unlimited
1
ICX7250-24 Router#
```

# show license unit

Displays general information about all software licenses on a device.

## Syntax

**show license unit** *unit_id* [ **index** *index_number* ]

## Parameters

*unit_id*

Indicates the unit ID number. The *unit_id* can be from 1 through 12.

**index** *index_number*

Specifies the software license file for a specific stack.

## Modes

Privileged EXEC level.

## Usage Guidelines

The command can be used to display software licensing information for both SAU and node-locked/non-node-locked XML licenses for a specified unit on a device.

## Command Output

The **show license unit** command displays the following information:

| Output field | Description |
|---|---|
| Unit | Unit number assigned in the stack. For standalone units, the unit number is 1. |
| License Name | Name of Software Authenticated Upgrade (SAU) license installed. |
| L3 Premium | (Yes, No) Indicates whether Layer 3 features are enabled by the license. |
| Port Speed Upgrade | (Yes, No) Indicates whether the license allows ports to be upgraded from the default speed (1 Gbps). |
| Speed | Speed to which ports covered by the license can be upgraded (10 Gbps). |
| Ports | Indicates the number of ports covered by the license. |
| MACsec | Indicates whether a MACsec license is installed. |
| Serial # | License serial number. |
| Index | The index number specifies the software license file for a specific stack. The index number is generated by the member unit. |
| Lid | The license ID. This number is embedded in the Ruckus device. |

| Output field | Description |
|---|---|
| Lic Mode | Indicates whether the license is a non-node-locked license or node-locked license. |
| License name | The name of the XML license installed for the license index number on the stack unit. |
| Lid/Serial No | The license ID. The number is embedded in the Ruckus device. The serial number for only a non-node locked license. The serial number is generated when you request a license through the license portal. The serial number is not the device name. |
| License Type | Indicates whether the license is normal (permanent) or trial (temporary). |
| Status | Indicates the status of the license:<br>• Valid - A license is valid if the LID matches the license ID of the device for which the license was purchased, and the package name is recognized by the system.<br>• Invalid - The LID does not match the license ID of the device for which the license was purchased.<br>• Active - The license is valid and in effect on the device.<br>• Not used - The license is not in effect on the device.<br>• Expired - For trial licenses only, this indicates that the trial license has expired.<br>• Duplicated - For non-node-locked licenses, this indicates that the same serial number is used for devices in a stacking system. |
| License Period | If the license type is trial (temporary), this field displays the number of days the license is valid. If the license type is normal (permanent), this field displays Unlimited. |
| License Capacity | The port capacity of the Ports of Demand (PoD) license. |

# Examples

The following **show license unit** command output displays information about the SAU licenses and XML licenses on the device. Hardware license information is not displayed.

```
ICX7250-24 Router# show license unit 1
Unit License Name    L3 Premium   Port Speed Upgrade  Speed    Ports  MACSec  Serial# (Prem/PoD/MACsec)
1     L3-PREM-2X10G   Yes         Yes                 10G      2      NA      PR320400289/NA/NA

Index    Lic Mode        Lic Name              Lid/Serial No  Lic Type    Status     Lic Period    Lic
Capacity
Stack unit 1:
1        Non-Node Lock   ICX7250-PREM-LIC-SW    EN0E583FD98    Normal      Active     Unlimited
1
2        Node Lock       ICX7250-10G-LIC-POD    fwjINHGnFMF    Normal      Active     Unlimited
2
ICX7250-24 Router#
```

# show link-error-disable

Displays the ports that are enabled with the port flap dampening feature.

## Syntax

**show link-error-disable** [ **all** ]

## Parameters

**all**

Displays all ports with the port flap dampening feature enabled.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

Ports that have been disabled due to the port flap dampening feature are identified in the output of the **show link-error-disable** command.

## Command Output

The **show link-error-disable** command displays the following information.

| Output field | Description |
|---|---|
| Port | Specifies the port number. |
| threshold | The number of times that the port link state goes from up to down and down to up before the wait period is activated. |
| sampling_period | The number of seconds during which the specified toggle threshold can occur before the wait period is activated. |
| waiting_period | The number of seconds during which the port remains disabled (down) before it becomes enabled. |

# Examples

The following is sample output from the **show link-error-disable all** command.

```
device# show link-error-disable all

Port1/1/1 is configured for link-error-disable
            threshold:1, sampling_period:10, waiting_period:0
Port1/1/2 is configured for link-error-disable
            threshold:1, sampling_period:10, waiting_period:0
Port1/1/3 is configured for link-error-disable
            threshold:1, sampling_period:10, waiting_period:0
Port1/1/4 is configured for link-error-disable
            threshold:1, sampling_period:10, waiting_period:0
Port1/1/5 is configured for link-error-disable
            threshold:4, sampling_period:10, waiting_period:2
Port1/1/9 is configured for link-error-disable
            threshold:2, sampling_period:20, waiting_period:0
```

# show link-keepalive

Displays the UDLD information.

## Syntax

**show link-keepalive** [ **ethernet** *stackid/slot/port* ]

## Parameters

**ethernet** *stackid/slot/port*

Displays UDLD information for the specified Ethernet port.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show link-keepalive** command displays the following information:

| Output field | Description |
|---|---|
| Total link-keepalive enabled ports | The total number of ports on which UDLD is enabled. |
| Keepalive Retries | The number of times a port will attempt the health-check before concluding that the link is down. |
| Keepalive Interval | The number of seconds between health check packets. |
| Port | The port number. |
| Physical Link | The state of the physical link. This is the link between the Ruckus port and the directly connected device. |
| Logical Link | The state of the logical link. This is the state of the link between this port and the port on the other end of the link. |
| State | The traffic state of the port. |
| Link-vlan | The ID of the tagged VLAN in the UDLD packet. |

The **show link-keepalive ethernet** command displays the following information:

| Output field | Description |
|---|---|
| Current State | The state of the logical link. This is the link between this port and the port on the other end of the link. |
| Remote MAC Addr | The MAC address of the port or device at the remote end of the logical link. |
| Local Port | The port number on this device. |
| Remote Port | The port number on the device at the remote end of the link. |
| Local System ID | A unique value that identifies this device. The ID can be used by Ruckus technical support for troubleshooting. |

| Output field | Description |
|---|---|
| Remote System ID | A unique value that identifies the device at the remote end of the link. |
| Packets sent | The number of UDLD health-check packets sent on this port. |
| Packets received | The number of UDLD health-check packets received on this port. |
| Transitions | The number of times the logical link state has changed between up and down. |
| Port blocking | Information used by Ruckus technical support for troubleshooting. |
| Link-vlan | The ID of the tagged VLAN in the UDLD packet. |
| BM disabled | Information used by Ruckus technical support for troubleshooting. |

# Examples

The following example shows the UDLD information for all ports.

```
device# show link-keepalive
Total link-keepalive enabled ports: 4
Keepalive Retries: 3    Keepalive Interval: 1 Sec.
Port    Physical Link   Logical Link    State           Link-vlan
1/1/1     up              up            FORWARDING      3
1/1/2     up              up            FORWARDING
1/1/3     down            down          DISABLED
1/1/4     up              down          DISABLED
```

The following example show the UDLD information for a specific port.

```
device# show link-keepalive ethernet 1/4/1
Current State   : up            Remote MAC Addr  : 0000.00d2.5100
Local Port      : 1/4/1         Remote Port      : 1/2/1
Local System ID : e0927400      Remote System ID : e0d25100
Packets sent    : 254           Packets received : 255
Transitions     : 1             Link-vlan        : 100
```

# show link-oam info

Displays the OAM information on EFM-OAM-enabled ports.

## Syntax

**show link-oam info** [ **detail** [ **ethernet** *stackid/slot/port* [ [ **to** *stackid/slot/port* ] [ **ethernet** *stackid/slot/port* ]... ] ] ]

## Parameters

**detail**

Displays detailed EFM-OAM information.

**ethernet**

Displays the detailed EFM-OAM information for a specific Ethernet interface.

*stackid/slot/port*

Specifies the interface details.

**to**

Configures a range of interfaces.

## Modes

Privileged EXEC mode

Global configuration mode

EFM-OAM protocol configuration mode

## Command Output

The **show link-oam info** command displays the following information:

| Output field | Description |
|---|---|
| Ethernet | Displays the interface details |
| Link Status | Displays the status of the link (up or down) |
| OAM Status | Displays the status of OAM |
| Mode | Displays the operational mode of EFM-OAM |
| Local Stable | Displays the local OAM status |
| Remote Stable | Displays the remote OAM status |
| multiplexer action | Displays the local/remote multiplexer action |
| parse action | Displays the local/remote parse action |
| stable | Displays the local/remote OAM status |
| state | Displays the local/remote EFM-OAM state |
| loopback support | Indicates whether there is support for loopback for remote/local |
| dying-gasp | Indicates whether there is support for dying gasp for remote/local |
| critical-event | Indicates whether there is support for critical-event for remote/local |

| Output field | Description |
|---|---|
| link-fault | Indicates whether there is support for link-fault for remote/local |

# Examples

The following example displays the OAM information on all EFM-OAM-enabled ports.

```
device(config)# show link-oam info
Ethernet Link Status     OAM Status      Mode       Local Stable    Remote Stable
1/1/1    up              up              active     satisfied       satisfied
1/1/2    up              up              passive    satisfied       satisfied
1/1/3    up              up              active     satisfied       satisfied
1/1/4    up              init            passive    unsatisfied     unsatisfied
1/1/5    down            down            passive    unsatisfied     unsatisfied
1/1/6    down            down            passive    unsatisfied     unsatisfied
1/1/7    down            down            passive    unsatisfied     unsatisfied
```

The following example displays detailed EFM-OAM information on all EFM-OAM-enabled ports.

```
device(config)# show link-oam info detail
OAM information for Ethernet port: 10/1/1
        +link-oam mode:        passive
        +link status:          down
        +oam status:           down
        Local information
                multiplexer action:  forward
                parse action:        forward
                stable:              unsatisfied
                state:               linkFault
                loopback state:      disabled
                dying-gasp:          false
                critical-event:      false
                link-fault:          true
        Remote information
                multiplexer action:  forward
                parse action:        forward
                stable:              unsatisfied
                loopback support:    disabled
                dying-gasp:          false
                critical-event:      true
                link-fault:          false

OAM information for Ethernet port: 10/1/3
        +link-oam mode:        active
        +link status:          up
        +oam status:           down
        Local information
                multiplexer action:  forward
                parse action:        forward
                stable:              unsatisfied
                state:               activeSend
                loopback state:      disabled
                dying-gasp:          false
                critical-event:      false
                link-fault:          false
        Remote information
                multiplexer action:  forward
                parse action:        forward
                stable:              unsatisfied
                loopback support:    disabled
                dying-gasp:          false
                critical-event:      false
                link-fault:          false

OAM information for Ethernet port: 10/1/4
        +link-oam mode:        active
        +link status:          up
        +oam status:           up
        Local information
                multiplexer action:  forward
                parse action:        forward
                stable:              satisfied
                state:               up
                loopback state:      disabled
                dying-gasp:          false
                critical-event:      false
                link-fault:          false
        Remote information
                multiplexer action:  forward
                parse action:        forward
                stable:              satisfied
                loopback support:    disabled
                dying-gasp:          false
                critical-event:      true
                link-fault:          false
```

The following example displays detailed EFM-OAM information on a range of EFM-OAM-enabled ports.

```
device(config)# show link-oam info detail ethernet 1/1/3 to 1/1/8
OAM information for Ethernet port: 1/1/3
        +link-oam mode:        active
        +link status:          up
        +oam status:           up
        Local information
                multiplexer action:  forward
                parse action:        forward
                stable:              satisfied
                state:               up
                loopback state:      disabled
                dying-gasp:          false
                critical-event:      false
                link-fault:          false
        Remote information
                multiplexer action:  forward
                parse action:        forward
                stable:              satisfied
                loopback support:    disabled
                dying-gasp:          false
                critical-event:      false
                link-fault:          false

Link OAM is not enabled on port 1/1/4
Link OAM is not enabled on port 1/1/5
Link OAM is not enabled on port 1/1/6
Link OAM is not enabled on port 1/1/7
Link OAM is not enabled on port 1/1/8
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show link-oam statistics

Displays the OAM statistics of OAM-enabled ports.

## Syntax

**show link-oam statistics** [ **detail** [ **ethernet** *stackid/slot/port* [ [ **to** *stackid/slot/port* ] [ **ethernet** *stackid/slot/port* ]... ] ] ]

## Parameters

**detail**

Displays detailed EFM-OAM statistics.

**ethernet**

Displays the detailed EFM-OAM statistics of a specific ethernet interface.

*stackid/slot/port*

Specifies the interface details.

**to**

Configures a range of interfaces.

## Modes

Privileged EXEC mode

Global configuration mode

EFM-OAM protocol configuration mode

## Command Output

The **show link-oam statistics** command displays the following information:

| Output field | Description |
|---|---|
| Tx PDUs | Displays the number of PDUs transmitted |
| Rx PDUs | Displays the number of PDUs received |
| information OAMPDUs | Displays the number of information OAMPDUs transmitted/received |
| loopback control OAMPDUs | Displays the number of loopback control OAMPDUs transmitted/received |
| variable request OAMPDUs | Displays the number of variable request OAMPDUs transmitted/received |
| variable response OAMPDUs | Displays the number of variable response OAMPDUs transmitted/received |
| unique event notification OAMPDUs | Displays the number of unique event notification OAMPDUs transmitted/received |
| duplicate event notification OAMPDUs | Displays the number of duplicate event notification OAMPDUs transmitted/received |
| organization specific OAMPDUs | Displays the number of organization specific OAMPDUs transmitted/received |
| link-fault records | Displays the number of link-fault records transmitted/received |
| critical-event records | Displays the number of critical-event records transmitted/received |
| dying-gasp records | Displays the number of dying-gasp records transmitted/received |
| loopback control OAMPDUs dropped | Displays the number of dropped loopback control OAMPDUs |

| Output field | Description |
|---|---|
| unsupported OAMPDUs | Displays the number of unsupported OAMPDUs |
| discarded TLVs | Displays the number of discarded TLVs |
| unrecognized TLVs | Displays the number of unrecognized TLVs |

# Examples

The following example displays the OAM statistics on all EFM-OAM-enabled ports.

```
device(config)# show link-oam statistics
Ethernet Tx Pdus         Rx Pdus
10/1/1   377908         377967
10/1/3   400            44
10/1/4   400            385
10/1/5   400            385
10/1/6   400            385
```

The following example displays detailed EFM-OAM statistics on all EFM-OAM-enabled ports.

```
device(config)# show link-oam statistics detail
OAM statistics for Ethernet port: 10/1/1
        Tx statistics
                information OAMPDUs:                    377908
                loopback control OAMPDUs:               0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                link-fault records:                     0
                critical-event records:                 0
                dying-gasp records:                     0
        Rx statistics
                information OAMPDUs:                    377967
                loopback control OAMPDUs:               0
                loopback control OAMPDUs dropped:       0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                unsupported OAMPDUs:                    0
                link-fault records:                     0
                critical-event records:                 377395
                dying-gasp records:                     0
                discarded TLVs:                         0
                unrecognized TLVs:                      0

OAM statistics for Ethernet port: 10/1/3
        Tx statistics
                information OAMPDUs:                    427
                loopback control OAMPDUs:               0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                link-fault records:                     0
                critical-event records:                 0
                dying-gasp records:                     0
        Rx statistics
                information OAMPDUs:                    44
                loopback control OAMPDUs:               0
                loopback control OAMPDUs dropped:       0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                unsupported OAMPDUs:                    0
                link-fault records:                     0
                critical-event records:                 0
                dying-gasp records:                     0
                discarded TLVs:                         0
                unrecognized TLVs:                      0

OAM statistics for Ethernet port: 10/1/4
        Tx statistics
                information OAMPDUs:                    428
                loopback control OAMPDUs:               0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                link-fault records:                     0
                critical-event records:                 0
                dying-gasp records:                     0
```

```
        Rx statistics
                information OAMPDUs:                    413
                loopback control OAMPDUs:               0
                loopback control OAMPDUs dropped:       0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                unsupported OAMPDUs:                    0
                link-fault records:                     0
                critical-event records:                 350
                dying-gasp records:                     0
                discarded TLVs:                         0
                unrecognized TLVs:                      0
```

The following example displays detailed EFM-OAM statistics on a range of EFM-OAM-enabled ports.

```
device(config)# show link-oam statistics detail ethernet 1/1/3 to 1/1/8
OAM statistics for Ethernet port: 1/1/3
        Tx statistics
                information OAMPDUs:                    255390
                loopback control OAMPDUs:               0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                link-fault records:                     0
                critical-event records:                 0
                dying-gasp records:                     0
        Rx statistics
                information OAMPDUs:                    282796
                loopback control OAMPDUs:               0
                loopback control OAMPDUs dropped:       0
                variable request OAMPDUs:               0
                variable response OAMPDUs:              0
                unique event notification OAMPDUs:      0
                duplicate event notification OAMPDUs:   0
                organization specific OAMPDUs:          0
                unsupported OAMPDUs:                    0
                link-fault records:                     0
                critical-event records:                 0
                dying-gasp records:                     0
                discarded TLVs:                         0
                unrecognized TLVs:                      0

Link OAM is not enabled on port 1/1/4
Link OAM is not enabled on port 1/1/5
Link OAM is not enabled on port 1/1/6
Link OAM is not enabled on port 1/1/7
Link OAM is not enabled on port 1/1/8
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show lldp

Displays a summary of the Link Layer Discovery Protocol (LLDP) configuration settings.

## Syntax

**show lldp**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show lldp** command displays the following information.

| Output field | Description |
|---|---|
| LLDP transmit interval | The number of seconds between regular LLDP packet transmissions. |
| LLDP transmit hold multiplier | The multiplier used to compute the actual time-to-live (TTL) value of an LLDP advertisement. The TTL value is the transmit interval multiplied by the transmit hold multiplier. |
| LLDP transmit delay | The number of seconds that the LLDP agent will wait after transmitting an LLDP frame before transmitting another LLDP frame. |
| LLDP SNMP notification interval | The number of seconds between transmission of SNMP LLDP traps (lldpRemTablesChange) and SNMP LLDP-MED traps (lldpXMedTopologyChangeDetected). |
| LLDP reinitialize delay | The minimum number of seconds that the device will wait from when LLDP is disabled on a port, until a request to re-enable LLDP on that port is honored. |
| LLDP-MED fast start repeat count | The number of seconds between LLDP frame transmissions when an LLDP-MED endpoint is newly detected. |
| LLDP maximum neighbors | The maximum number of LLDP neighbors for which LLDP data will be retained, per device. |
| LLDP maximum neighbors per port | The maximum number of LLDP neighbors for which LLDP data will be retained, per port. |

## Examples

The following is sample output from the **show lldp** command.

```
device# show lldp

LLDP transmit interval             : 10 seconds
LLDP transmit hold multiplier    : 4 (transmit TTL: 40 seconds)
LLDP transmit delay              : 1 seconds
LLDP SNMP notification interval  : 5 seconds
LLDP reinitialize delay          : 1 seconds
LLDP-MED fast start repeat count : 3
LLDP maximum neighbors              : 392
LLDP maximum neighbors per port  : 4
```

## Related Commands

show lldp local-info, show lldp neighbors, show lldp statistics

# show lldp local-info

Displays the details of the Link Layer Discovery Protocol (LLDP) advertisements that will be transmitted on each port.

## Syntax

**show lldp local-info** *ports* { *all* | *ethernet stack-id/slot/port* [ *to stack-id/slot/port* | [*ethernet stack-id/slot/port to stack-id/slot/port* | *ethernet stack-id/slot/port* ] ... ]}

## Parameters

*ports*
: Displays the details of the LLDP advertisements that will be transmitted on the specified port.

*all*
: Displays the details of the LLDP advertisements that will be transmitted on all LLDP-enabled ports.

*ethernet stack-id/slot/port*
: Displays the details of the LLDP advertisements that will be transmitted on the specified Ethernet port.

*to stack- id/slot/port*
: Displays the details of the LLDP advertisements that will be transmitted on a range of ports.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

The contents of the show output will vary depending on which Threshold Limit Values (TLVs) are configured to be advertised.

If you do not specify any ports or use the **all** keyword, by default, the report shows the local information advertisements for all ports.

# Examples

The following is a sample output of the **show lldp local-info** command.

```
device# show lldp local-info

Local port: 1/1/9:1
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.3294
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
  + Port description    : "10GigabitEthernet1/1/9:1"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation supported, but disabled
    Operational MAU type   : Other
  + Link aggregation: aggregated (aggregated port ifIndex: 21)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

Local port: 1/1/9:2
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.3295
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
  + Port description    : "10GigabitEthernet1/1/9:2"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation not supported
    Operational MAU type   : 77
  + Link aggregation: aggregated (aggregated port ifIndex: 21)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

Local port: 1/1/9:3
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.3296
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
  + Port description    : "10GigabitEthernet1/1/9:3"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation not supported
    Operational MAU type   : 162
  + Link aggregation: aggregated (aggregated port ifIndex: 21)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

Local port: 1/1/9:4
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.3297
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
  + Port description    : "10GigabitEthernet1/1/9:4"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation not supported
    Operational MAU type   : b10G1GbasePRXD1
  + Link aggregation: aggregated (aggregated port ifIndex: 21)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

Local port: 1/1/11:1
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.329c
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
```

```
  + Port description    : "10GigabitEthernet1/1/11:1"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation supported, but disabled
    Operational MAU type   : Other
  + Link aggregation: aggregated (aggregated port ifIndex: 29)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

Local port: 1/1/11:2
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.329d
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
  + Port description    : "10GigabitEthernet1/1/11:2"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation not supported
    Operational MAU type   : 162
  + Link aggregation: aggregated (aggregated port ifIndex: 29)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

Local port: 1/1/11:3
  + Chassis ID (MAC address): 0000.0043.4343
  + Port ID (MAC address): cc4e.2438.329e
  + Time to live: 120 seconds
  + System name         : "775026Q-Seth"
  + Port description    : "10GigabitEthernet1/1/11:3"
  + System capabilities : bridge, router
    Enabled capabilities: bridge, router
  + 802.3 MAC/PHY          : auto-negotiation not supported
    Operational MAU type   : b10G1GbasePRXD1
  + Link aggregation: aggregated (aggregated port ifIndex: 29)
  + Maximum frame size: 10200 octets
  + Port VLAN ID: none
  + Management address (IPv4): 10.37.160.43

<<output truncated>>
```

# show lldp neighbors

Displays a list of current LLDP neighbors and details of the latest advertisements received from Link Layer Discovery Protocol (LLDP) neighbors.

## Syntax

**show lldp neighbors** [ **detail ports** { **all** | **ethernet** *stack-id/slot/port* [ **to** *stack-id/slot/port* | [ **ethernet** *stack-id/slot/port* **to** *stack-id/slot/port* | **ethernet** *stack-id/slot/port* ] ... ] } ]

## Parameters

**detail**

Displays detailed neighbor data.

**ports**

Displays the details of the latest advertisements received from LLDP neighbors for the specified port.

**all**

Displays the details of the latest advertisements received from LLDP neighbors for all LLDP-enabled ports.

**ethernet** *stack-id/slot/port*

Displays the details of the latest advertisements received from LLDP neighbors for the specified Ethernet port.

**to** *stack-id/slot/port*

Displays the details of the latest advertisements received from LLDP neighbors for a range of ports.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show lldp neighbors** command displays the following information.

| Output field | Description |
|---|---|
| Lcl Port | The local LLDP port number. |
| Chassis ID | The identifier for the chassis. Ruckus ICX devices use the base MAC address of the device as the Chassis ID. |
| Port ID | The identifier for the port. Ruckus ICX devices use the permanent MAC address associated with the port as the port ID. |
| Port Description | The description for the port. Ruckus ICX devices use the ifDescr MIB object from MIB-II as the port description. |
| System Name | The administratively-assigned name for the system. Ruckus ICX devices use the sysName MIB object from MIB-II, which corresponds to the CLI **hostname** command setting. |

# Examples

The following is sample output from the **show lldp neighbors** command.

```
device# show lldp neighbors

Lcl Port Chassis ID      Port ID         Port Description            System Name
1/1/9:1  0000.0126.2057  748e.f8f9.7489  10GigabitEthernet1/1/10     7750Stk
1/1/9:2  0000.0126.2057  748e.f8f9.7509  10GigabitEthernet2/1/10     7750Stk
1/1/9:3  0000.0126.2057  748e.f8f9.7488  10GigabitEthernet1/1/9      7750Stk
1/1/9:4  0000.0126.2057  748e.f8f9.7508  10GigabitEthernet2/1/9      7750Stk
1/1/11:1 0000.4690.5353  cc4e.246c.e5a2  10GigabitEthernet1/2/2      7450Stk
1/1/11:2 0000.4690.5353  cc4e.246c.ea41  10GigabitEthernet2/2/1      7450Stk
1/1/11:3 0000.4690.5353  cc4e.246c.e5a1  10GigabitEthernet1/2/1      7450Stk
1/1/11:4 0000.4690.5353  cc4e.246c.df21  10GigabitEthernet3/2/1      7450Stk
```

The following is sample output from the **show lldp neighbors detail** command.

```
device# show lldp neighbors detail ports ethernet 1/1/9:1

Local port: 1/1/9:1
Neighbor  : 748e.f8f9.7489, TTL 92 seconds
    + Chassis ID (MAC address) : 0000.0126.2057
    + Port ID (MAC address)    : 748e.f8f9.7489
    + Time to live             : 120 seconds
    + System name              : "7750Stk-Seth"
    + Port description         : "10GigabitEthernet1/1/10"
    + System capabilities      : bridge, router
      Enabled capabilities     : bridge, router
    + 802.3 MAC/PHY            : auto-negotiation supported, but disabled
      Operational MAU type     : Other
    + Link aggregation         : aggregated (aggregated port ifIndex: 10)
    + Maximum frame size       : 10200 octets
    + Port VLAN ID             : none
    + Management address (IPv4): 10.37.160.126
```

# show lldp statistics

Displays Link Layer Discovery Protocol (LLDP) global and per-port statistics.

## Syntax

**show lldp statistics**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

# Examples

The following is sample output from the **show lldp statistics** command.

```
device# show lldp statistics

Last neighbor change time: 3 hour(s) 37 minute(s) 59 second(s) ago

Neighbor entries added          : 25
Neighbor entries deleted        : 17
Neighbor entries aged out       : 3
Neighbor advertisements dropped : 0

Port       Tx Pkts   Rx Pkts   Rx Pkts   Rx Pkts   Rx TLVs   Rx TLVs Neighbors
             Total     Total  w/Errors Discarded Unrecognz Discarded  Aged Out
1/1/1            0         0         0         0         0         0         0
1/1/2            0         0         0         0         0         0         0
1/1/3            0         0         0         0         0         0         0
1/1/4            0         0         0         0         0         0         0
1/1/5            0         0         0         0         0         0         0
1/1/6            0         0         0         0         0         0         0
1/1/7            0         0         0         0         0         0         0
1/1/8            0         0         0         0         0         0         0
1/1/9:1        523       522         0         0         0         0         0
1/1/9:2        475       476         0         0         0         0         1
1/1/9:3        476       476         0         0         0         0         1
1/1/9:4        475       477         0         0         0         0         1
1/1/10           0         0         0         0         0         0         0
1/1/11:1       510       524         0         0         0         0         0
1/1/11:2       510       524         0         0         0         0         0
1/1/11:3       511       525         0         0         0         0         0
1/1/11:4       510       524         0         0         0         0         0
1/1/12           0         0         0         0         0         0         0
1/1/13           0         0         0         0         0         0         0
1/1/14           0         0         0         0         0         0         0
1/1/15           0         0         0         0         0         0         0
1/1/16           0         0         0         0         0         0         0
1/1/17           0         0         0         0         0         0         0
1/1/18           0         0         0         0         0         0         0
1/1/19           0         0         0         0         0         0         0
1/1/20           0         0         0         0         0         0         0
1/2/1            0         0         0         0         0         0         0
1/2/2            0         0         0         0         0         0         0
1/2/3            0         0         0         0         0         0         0
1/2/4            0         0         0         0         0         0         0
1/2/5            0         0         0         0         0         0         0
1/2/6            0         0         0         0         0         0         0
1/3/1            0         0         0         0         0         0         0
1/3/2            0         0         0         0         0         0         0
1/3/3            0         0         0         0         0         0         0
1/3/4            0         0         0         0         0         0         0
1/3/5            0         0         0         0         0         0         0
1/3/6            0         0         0         0         0         0         0
```

# show local-userdb

Displays a list of local user databases configured on the device and the number of users in each database.

## Syntax

**show local-userdb** [ *db-name* [*user-name* ] ]

## Parameters

*db-name*

Displays information for the specified local user database. The database name and the username can be up to 31 characters.

*user-name*

Displays information for the specified user in the specified user database.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

Web Authentication configuration mode

## Examples

The following example displays the list of all local user databases and the number of users in each database.

```
device# show local-userdb
==========================================================================
Local User Database Name : My_Database
Number of users in the database : 4
==========================================================================
Local User Database Name : test
Number of users in the database : 3
==========================================================================
Local User Database Name : test123
Number of users in the database : 3
```

The following example displays the details of a particular user database. The passwords are encrypted in the example.

```
device#show local-userdb test
==========================================================================
Local User Database : test
Username           Password
--------           --------
user1              $e$&Z9'%*&+
user2              $e$,)A=)65N,%-3*%1?@U
user3              $e$5%&-5%YO&&A1%6%<@U
```

The following example displays details of a particular user in a specific database.

```
device# show local-userdb db1 user1
Username = user1  Password = $e$%U*V
```

# show logging

Displays the Syslog messages in the device local buffer.

## Syntax

**show logging**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show logging** command displays the following information.

| Output field | Description |
|---|---|
| Syslog logging | The state (enabled or disabled) of the Syslog buffer. |
| messages dropped | The number of Syslog messages dropped due to user-configured filters. By default, the software logs messages for all Syslog levels. You can disable individual Syslog levels, in which case the software filters out messages at those levels. Each time the software filters out a Syslog message, this counter is incremented. |
| flushes | The number of times the Syslog buffer has been cleared by the clear logging command or equivalent Web Management Interface option. |
| overruns | The number of times the dynamic log buffer has filled up and been cleared to hold new entries. For example, if the buffer is set for 100 entries, the 101st entry causes an overrun. After that, the 201st entry causes a second overrun. |
| level | The message levels that are enabled. Each letter represents a message type and is identified by the key (level code) below the value. If you disable logging of a message level, the code for that level is not listed. |
| messages logged | The total number of messages that have been logged since the software was loaded. |
| level code | The message levels represented by the one-letter codes. |

# Examples

The following is sample output from the **show logging** command.

```
device# show logging

Syslog logging: enabled (0 messages dropped, 0 flushes, 24 overruns)
    Buffer logging: level ACDMEINW, 50 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
Jan  1 00:00:47:I:System: Stack unit 1 PSU fan direction mismatch

Dynamic Log Buffer (50 lines):
Jan  3 20:23:10:I:Security: startup-config was changed by operator from console
Jan  3 20:17:25:I:Security: startup-config was changed by operator from console
Jan  3 19:50:43:I:MSTP: MST 1 Port 1/1/11:1  - Bridge TC Event
Jan  3 19:50:43:I:MSTP: MST 0 Port 1/1/11:1  - Bridge TC Event
Jan  3 19:49:29:I:System: Logical link on dynamic lag interface ethernet 1/1/9:3 is up.
Jan  3 19:49:29:I:System: Interface ethernet 1/1/9:3, state up
Jan  3 19:49:29:I:System: Logical link on dynamic lag interface ethernet 1/1/9:1 is up.
Jan  3 19:49:29:I:System: Interface ethernet 1/1/9:1, state up
Jan  3 19:49:29:I:MRP: Interface ethernet 1/1/9:1 of ring 51  Vlan 51, changing to forwarding
Jan  3 19:49:29:I:MSTP: MST 1 Port 1/1/11:1  - Bridge TC Event
Jan  3 19:49:29:I:MSTP: MST 1 Port 1/1/9:1  - Bridge TC Event
Jan  3 19:49:29:I:MSTP: MST 1 Port 1/1/9:1 - FORWARDING
Jan  3 19:49:29:I:MSTP: MST 1 Port 1/1/9:1 - LEARNING
Jan  3 19:49:29:I:MSTP: MST 0 Port 1/1/11:1  - Bridge TC Event
Jan  3 19:49:29:I:MSTP: MST 0 Port 1/1/9:1  - Bridge TC Event
Jan  3 19:49:29:I:MSTP: MST 0 Port 1/1/9:1 - FORWARDING
Jan  3 19:49:29:I:MSTP: MST 0 Port 1/1/9:1 - LEARNING
Jan  3 19:49:29:I:System: Logical link on dynamic lag interface ethernet 1/1/9:4 is up.
Jan  3 19:49:29:I:System: Interface ethernet 1/1/9:4, state up
Jan  3 19:49:29:I:System: Logical link on dynamic lag interface ethernet 1/1/9:2 is up.
Jan  3 19:49:29:I:System: Interface ethernet 1/1/9:2, state up
Jan  3 19:49:29:I:MRP: Interface ethernet 1/1/9:1 of ring 51  Vlan 51, changing to preforwarding
Jan  3 19:49:29:I:MSTP: MST 1 Port 1/1/9:1 - DISCARDING
Jan  3 19:49:29:I:MSTP: MST 0 Port 1/1/9:1 - DISCARDING
Jan  3 19:49:29:I:Trunk: Group (1/1/9:1, 1/1/9:2, 1/1/9:3, 1/1/9:4) created by 802.3ad link-aggregation
module.
Jan  3 19:49:12:I:System: Interface ethernet 1/1/9:2, state down
Jan  3 19:49:12:I:System: Logical link on dynamic lag interface ethernet 1/1/9:2 is down.
Jan  3 19:49:12:I:MRP: Interface ethernet 1/1/9:1 of ring 51  Vlan 51, changing to disabled
Jan  3 19:49:12:I:MSTP: MST 0 Port 1/1/9:1 - DISCARDING
Jan  3 19:49:12:I:MSTP: MST 1 Port 1/1/9:1 - DISCARDING
Jan  3 19:49:12:I:Trunk: Group (1/1/9:1, 1/1/9:2, 1/1/9:3, 1/1/9:4) removed by 802.3ad link-aggregation
module.
Jan  3 19:49:12:I:System: Interface ethernet 1/1/9:4, state down
Jan  3 19:49:12:I:System: Logical link on dynamic lag interface ethernet 1/1/9:4 is down.
Jan  3 19:49:12:I:System: Interface ethernet 1/1/9:1, state down
Jan  3 19:49:12:I:System: Logical link on dynamic lag interface ethernet 1/1/9:1 is down.
Jan  3 19:49:12:I:System: Interface ethernet 1/1/9:3, state down
Jan  3 19:49:12:I:System: Logical link on dynamic lag interface ethernet 1/1/9:3 is down.
```

# show loop-detection resource

Displays the hardware and software resource information about loop detection.

## Syntax

**show loop-detection resource**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show loop-detection resource** command displays the following information.

| Output field | Description |
|---|---|
| alloc | Memory allocated |
| in-use | Memory in use |
| avail | Available memory |
| get-fail | The number of get requests that have failed |
| limit | The maximum memory allocation |
| get-mem | The number of get-memory requests |
| size | The size of the memory |
| init | The number of requests initiated |

## Examples

The following is sample output from the **show loop-detection resource** command.

```
device# show loop-detection resource

Vlans configured loop-detection use 1 HW MAC
Vlans not configured but use HW MAC: 1 10
alloc                    in-use  avail    get-fail  limit   get-mem   size   init
configuration pool 16     6         10       0           3712     6         15          16
linklist pool       16      10        6          0        3712        10        16      16
```

## Related Commands

show loop-detection status

# show loop-detection status

Displays loop detection status.

## Syntax

**show loop-detection status**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show loop-detection status** command. If a port is disabled in Strict mode, it shows "ERR-DISABLE by itself." If it is disabled due to its associated VLAN, it shows "ERR-DISABLE by vlan<num>."

```
device# show loop-detection status

loop detection packets interval: 10 (unit 0.1 sec)
Number of err-disabled ports: 3
You can re-enable err-disable ports one by one by "disable" then "enable"
under interface config, re-enable all by "clear loop-detect", or
configure "errdisable recovery cause loop-detection" for automatic recovery

index     port/vlan     status                                #errdis    sent-pkts    recv-pkts
1            1/1/13           untag, LEARNING              0            0                    0
2            1/1/15           untag, BLOCKING              0            0                    0
3            1/1/17           untag, DISABLED              0            0                    0
4         1/1/18         ERR-DISABLE by itself     1            6                 1
5            1/1/19           ERR-DISABLE by vlan12     0               0                    0
6            vlan12     ERR-DISABLE ports         2            24                2
```

# show loop-detect no-shutdown-status

Shows the status of interfaces in a loop.

## Syntax

**show loop-detect no-shutdown-status**

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show loop-detect no-shutdown-status** command displays the following information:

| Output field | Description |
|---|---|
| Port | The specific interface |
| Loop status | The duration the port has been in a loop |

## Examples

The following example shows the ports and their loop statuses.

```
device# show loop-detection no-shutdown-status

loop detection no shutdown syslog interval : 5     (unit 1 min /Default 5 min)
loop detection no shutdown port status     :
Note: Port's loop status gets cleared if loop is not detected in a particular interval window

                Port    ||  Loop Status
      ==================||=========================
       ethernet  1/1/7  ||  (In Loop For 2309 Seconds)
       ethernet  1/1/15 ||  (In Loop For 2309 Seconds
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show lrm-adapter ethernet

Displays the LRM adapter parameters.

## Syntax

**show lrm-adapter ethernet**

## Modes

Privileged EXEC mode

## Usage Guidelines

The command is available only for ICX7750-48F 10G access ports.

## Command Output

The **show lrm-adapter ethernet** command displays the following information:

| Output field | Description |
|---|---|
| MCU Firmware Version | Firmware version of the Micro Controller Unit (MCU). |
| Power Mode | Power mode of LRM adapter (high/low). |
| Vendor | Vendor name. |
| Vendor PN | Vendor part number. |
| Vendor SN | Vendor serial number. |

## Examples

The following example displays the LRM parameters on ethernet 1/2/4.

```
device#sh lrm-adapter ethernet 1/2/4
LRM Adapter on port:1/2/4
=========================
MCU Firmware Version:01.05
Power Mode: High
Vendor: RUCKUS
Vendor PN: 58000007401
Vendor SN: AAF2120900007U5
device#
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show l2protocol dot1q-tunnel

Displays Q-in-Q BPDU tunnel information.

## Syntax

**show l2protocol dot1q-tunnel** { **counters** [ *unit / slot / port* | *lag-id* ] | **port** { *unit / slot / port* | *lag-id* } | **summary** | **vlan** *vlan-id* }

## Parameters

**counters**
Displays tunnel counters for all interfaces.

*unit / slot / port*
Displays tunnel counters for a specific interface.

*lag-id*
Displays tunnel counters for a LAG virtual interface.

**port** *unit / slot / port* | *lag-id*
Displays Q-in-Q BPDU tunnel configuration details on an interface or on a LAG virtual interface.

**summary**
Displays a summary of all ports that have Q-in-Q BPDU tunnel configurations.

**vlan** *vlan-id*
Displays Q-in-Q PPDU tunnel information of all dot1q-tunnel-enabled interfaces that are part of a specified VLAN.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

# Examples

The following example displays tunnel counter for each protocol on an interface.

```
device(config)# show l2protocol dot1q-tunnel counters 3/1/1
BPDU Tunnel Counters for 3/1/1:
   STP: Rx(customer)=239514 Tx(customer)=243824 Drop(customer)=0 Forward(service)=239514 Forward
Fail(service)=0
  PVST: Rx(customer)=0 Tx(customer)=0 Drop(customer)=0 Forward(service)=0 Forward Fail(service)=0
  LACP: Rx(customer)=9198 Tx(customer)=8362 Drop(customer)=0 Forward(service)=9198 Forward
Fail(service)=0
  LLDP: Rx(customer)=8181 Tx(customer)=8138 Drop(customer)=0 Forward(service)=8181 Forward
Fail(service)=0
   CDP: Rx(customer)=0 Tx(customer)=0 Drop(customer)=0 Forward(service)=0 Forward Fail(service)=0
   All: Unknown Rx BPDU Drop(service)=0 Tx fail(customer) Drop=0
   All: SVLAN invalid for Rx BPDU (customer) Drop=0
   All: Tag error for Rx BPDU Drop=0
```

The following example displays a summary of all ports that have Q-in-Q BPDU tunnel configurations.

```
device(config)# show l2protocol dot1q-tunnel summary
  BPDU Tunnel original MAC disabled
  BPDU Tunnel MAC =0100.0ccd.cdd1
  BPDU Tunnel CoS =5

  STP Tunnel Ports:  3/1/1 3/1/2 lg1
  LACP Tunnel Ports:  3/1/1 3/1/2
  LLDP Tunnel Ports:  3/1/1 3/1/2 lg1
  CDP Tunnel Ports:  3/1/1 3/1/2 lg1
  Rate limit enabled Ports: None
```

The following example displays Q-in-Q BPDU tunnel configuration details on an interface.

```
device(config)# show l2protocol dot1q-tunnel port 3/1/2
 BPDU Tunnel enabled on 3/1/2 for following protocols
 Protocols: CDP LACP LLDP STP

 STP drop Threshold: 100 pkts/sec
 STP shutdown Threshold: 200 pkts/sec
 STP current Rx Rate: 0 pkts/sec
 STP last Rx Time: 1 second(s) ago

 LACP drop Threshold: Not enabled
 LACP shutdown Threshold: Not enabled
 LLDP drop Threshold: Not enabled
 LLDP shutdown Threshold: Not enabled
 CDP drop Threshold: Not enabled
 CDP shutdown Threshold: Not enabled
 All protocol drop drop Threshold: Not enabled
 All protocol shutdown Threshold: Not enabled
```

The following example displays Q-in-Q BPDU tunnel information of all dot1q-tunnel-enabled interfaces that are part of a specified VLAN.

```
device(config)# show l2p dot1q-tunnel vlan 100
BPDU Tunnel enabled on 3/1/1 for following protocols
 Protocols: CDP LACP LLDP STP

 STP drop Threshold: Not enabled
 STP shutdown Threshold: Not enabled
 LACP drop Threshold: Not enabled
 LACP shutdown Threshold: Not enabled
 LLDP drop Threshold: Not enabled
 LLDP shutdown Threshold: Not enabled
 CDP drop Threshold: Not enabled
 CDP shutdown Threshold: Not enabled
 All protocol drop drop Threshold: Not enabled
 All protocol shutdown Threshold: Not enabled
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show mac-address

Displays the MAC address table.

## Syntax

**show mac-address** [ **ethernet** *stack/slot/port* | **vlan** *vlan-id* ] [ *mac-address* [ *mac-address-mask* ] ]

**show mac-address** [ **all** | **session** | **statistics** ]

## Parameters

**ethernet** *stack/slot/port*
Displays information for the specific Ethernet port.

**vlan** *vlan-id*
Displays the MAC address for the specified VLAN ID.

*mac-address*
Displays the information for the specified Ethernet MAC address.

*mac-address-mask*
Displays the information for the specified Ethernet MAC address mask.

**all**
Displays MAC address of all ports including the blocked ports.

**session**
Displays the MAC address of the ports in the session.

**statistics**
Displays the MAC address statistics.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Cluster configuration mode

## Command Output

The **show mac-address** command displays the following information:

| Output field | Description |
|---|---|
| MAC-Address | The MAC address. |
| Type | Indicates whether the MAC entry is static or dynamic. A static entry is one you create using the **static-mac-address** command. A dynamic entry is one that is learned by the software from network traffic. |

# Usage Guidelines

The **show mac-address** command output does not include MAC addresses for management ports, because these ports do not support typical MAC learning and MAC-based forwarding.

# Examples

The following example displays sample output of the **show mac-address** command.

```
device# show mac-address
Total active entries from all ports = 3
Total static entries from all ports = 1
MAC-Address        Port        Type      VLAN
0000.0034.1234     1/1/15      Static    1
0000.0038.2f24     1/1/14      Dynamic   1
0000.0038.2f00     1/1/13      Dynamic   1
0000.0086.b159     1/1/10      Dynamic   1
```

The following example displays sample output of the **show mac-address** command for a VLAN.

```
device# show mac-address vlan 1 0000.0000.0001
Total active entries from all ports = 16
MAC-Address        Port     Type      Index
0000.0000.0001     1/1/1      Dynamic   NA
Present in following devices (at hw index) :-
0 (8196 )        4 (8196 )
```

The following example displays two MAC Addresses learned on the VXLAN tunnel with destination IP address 2.2.2.2.

```
ICX7750-48F Router# show mac-address
Total active entries from all ports = 4
MAC-Address      Port          Type       VLAN
000c.2900.0011   1/1/48        Dynamic    101
000c.2900.0022   VxL-2.2.2.2   Dynamic    101
000c.2900.0023   VxL-2.2.2.2   Dynamic    102
000c.2900.0012   1/1/48        Dynamic    102
```

The following example displays information for VLAN 101, in this case a VLAN that is part of a VXLAN segment. The MAC Address of the local access port to the VXLAN tunnel, port 1/1/48, and the MAC address for the remote end of the tunnel, identified by its IP address 2.2.2.2 and the prefix VxL-, are displayed.

```
ICX7750-48F Router# show mac-address vlan 101
Total active entries from VLAN 101 = 2
MAC-Address      Port          Type          VLAN
000c.2900.0011   1/1/48        Dynamic       101
000c.2900.0022   VxL-2.2.2.2   Dynamic       101
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | The following options are removed as they were supported only on FSX devices: **mdup-status**, **mdb**, **source-rbridge** *source-rbridgeid*, **client-rbridge** *client-rbridgeid*. |
| 08.0.70 | The command has been enhanced to display MAC addresses for extended VLANs in a VXLAN segment. |

# show mac-address cluster

Displays all the MAC address entries for a cluster.

## Syntax

**show mac-address cluster** { *cluster-name* | *cluster-id* } [ **vlan** *vlan-id* ] [ **client** [ *client-name* | *client-id* ] ] [ **local** | **remote** [ **exclude-interface** | **interface** ] ]

## Parameters

*cluster-name*
Displays the details for the cluster with the specified cluster name.

*cluster-id*
Displays the details for the cluster with the specified cluster ID.

**vlan** *vlan-id*
Displays the details for the VLAN with the specified VLAN ID.

**client**
Displays the details for the configured client.

*client-name*
Displays the details for the configured client with the specified client name.

*client-id*
Displays the details for the configured client with the specified client ID.

**local**
Displays the cluster local MAC address.

**remote**
Displays the cluster remote MAC address.

**exclude-interface**
Displays the MAC address of the remote cluster excluding the interface MAC address of the remote cluster.

**interface**
Displays the cluster remote interface MAC address.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Cluster configuration mode

# Usage Guidelines

The **exclude-interface** and **interface** keywords are available only with the **remote** option. They are not available when the **client** option or the **vlan** option is used. When the **vlan** option is used, you can specify only the client name and not the client ID.

# Examples

The following example shows the output of the **show mac-address cluster** command.

```
device# show mac-address cluster 1000
Total Cluster Enabled(CL+CR+CCL+CCR) MACs: 1
Total Cluster Local(CL) MACs: 1
CCL: Cluster Client Local CCR:Cluster Client Remote CL:Local CR:Remote
Total active entries from all ports = 1
Total static entries from all ports = 3
MAC-Address       Port    Type     Index   MCT-Type   VLAN
0000.0022.3333    1/1/1   Static   4254    CML        20
0000.0022.3333    1/1/3   Static   4254    CML        20
0000.0022.3333    1/1/13  Static   4254    CML        20
```

# show mac-address mdb

Displays information about the MAC database used in cluster configuration.

## Syntax

**show mac-address mdb** [ **source-rbridge** *rbridge-id* **client-rbridge** *client-rbridge-id* ]

## Parameters

**source-rbridge** *rbridge-id*

Displays information about MAC database corresponding to a particular source RBridge. The range is from 1 to 4095.

**client-rbridge** *client-rbridge-id*

Displays information about MAC database corresponding to a particular client RBridge. The range is from 1 to 4095.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

.

# show mac-authentication configuration

Displays the global or interface level MAC authentication configuration.

## Syntax

**show mac-authentication configuration** [ **all** | **stack-unit** *id* | **ethernet** *unit*/*slot*/*port* ]

## Parameters

**all**

Displays the MAC authentication configuration on all interfaces.

**ethernet** *unit*/*slot*/*port*

Displays the MAC authentication configuration for a specific interface.

**stack-unit** *id*

Displays the MAC authentication configuration for a specific stack unit.

## Modes

EXEC or Privileged EXEC mode

Global configuration mode

## Usage Guidelines

## Command Output

The **show mac-authentication configuration** command displays the following information.

| Output field | Description |
|---|---|
| Status | Displays if MAC authentication is enabled or disabled |
| Auth-order | The authentication order enabled on the device |
| Default VLAN | The default VLAN specified on the device |
| Restricted VLAN | The restricted VLAN specified on the device |
| Critical VLAN | The critical VLAN specified on the device |
| Action on Auth failure | The action to be taken on authentication failure |
| MAC Session Aging | The status of the MAC session aging |
| Filter Strict Security | The status of filter strict security |
| Re-authentication | The status of re-authentication |
| Dot1x Override | The status of dot1x override |
| Password Override | The status of password override |
| Password Format | The configured password format |
| Reauth-period | The re-authentication period specified in seconds |
| Session max sw-age | The maximum software age configured on the device |

| Output field | Description |
|---|---|
| Session max hw-age | The maximum hardware age configured on the device |

The **show mac-authentication configuration all** | **ethernet** *unit/slot/port* command displays the following information.

| Output field | Description |
|---|---|
| Auth Order | Displays the authentication order |
| Action on Auth failure | Displays the action to be taken on authentication failure |
| Action on Auth timeout | Displays the action to be taken on authentication timeout |
| Filter Strict Security | Displays if filter strict security is enabled or disabled |
| DoS Protection | Displays if DoS protection is enabled or disabled |
| Source-guard Protection | Displays if Source-Guard Protection is enabled or disabled |
| Aging | Displays if aging is enabled or disabled |
| Max-sessions | Displays the count of the maximum sessions |
| Ingress-filtering | Displays if ingress filtering is enabled or disabled |

# Examples

The following example displays the system level MAC authentication configuration.

```
device# show mac-authentication configuration

Status : Enabled
Auth Order : dot1x mac-auth
Default VLAN : 4
Restricted VLAN : Not configured
Critical VLAN : Not configured
Action on Auth failure : Block traffic
MAC Session Aging : Enabled
Filter Strict Security : Enabled
Re-authentication : Enabled
Dot1x Override : Disabled
Password Override : Disabled
Password Format : xxxx.xxxx.xxxx
Reauth-period : 600 seconds
Session max sw-age : 120 seconds
Session max hw-age : 70 seconds
```

The following example displays the MAC authentication configuration for port 1/1/15.

```
device# configure terminal
device(config)# show mac-authentication configuration 1/1/15

Port 1/1/15 Configuration:
Auth Order                      : dot1x mac-auth
Action on Auth failure          : Block traffic
Action on Auth timeout          : Treat as a failed authentication
Filter Strict Security          : Enabled
DoS Protection                  : Disabled (limit = 512)
Source-guard Protection         : Disabled
Aging                           : Enabled
Max-sessions                    : 32
Auth Filter List (Filter/VLAN)  : 1/2
```

The following example displays the MAC authentication information on all interfaces.

```
device# configure terminal
device(config)# show mac-authentication configuration all

Port 1/1/1 Configuration:
Auth Order                      : dot1x mac-auth
Action on Auth failure          : Block traffic
Action on Auth timeout          : Treat as a failed authentication
Filter Strict Security          : Enabled
DoS Protection                  : Disabled (limit = 512)
Source-guard Protection         : Disabled
Reauth-timeout                  : 60 seconds
Aging                           : Enabled
Max-sessions                    : 2

Port 1/1/3 Configuration:
Auth Order                      : dot1x mac-auth
Action on Auth failure          : Block traffic
Action on Auth timeout          : Treat as a failed authentication
Filter Strict Security          : Enabled
DoS Protection                  : Disabled (limit = 512)
Source-guard Protection         : Disabled
Reauth-timeout                  : 60 seconds
Aging                           : Enabled
Max-sessions                    : 2
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.70 | The command was modified to include the stack-id option. |

# show mac-authentication ip-acl

Shows the Layer 3 access lists (ACLs) for MAC authentication.

## Syntax

**show mac-authentication ip-acl** { **all** | **stack-unit** *id* | **ethernet** *unit*/*slot*/*port* }

## Parameters

**all**

Specifies the ACLs at the global level.

**ethernet** *unit*/*slot*/*port*

Specifies the ACLs at the interface level.

**stack-unit** *id*

Displays MAC authentication ACLs for the specified stack unit.

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show mac-authentication ip-acl** command displays the following information.

| Output field | Description |
|---|---|
| Port | The port number. |
| MAC Addr | The MAC address of the client. |
| Inbound IPv4 ACL | The IPv4 ACL applied to the authenticated port in the inbound direction. |
| Outbound IPv4 ACL | The IPv4 ACL applied to the authenticated port in the outbound direction. |
| Inbound IPv6 ACL | The IPv6 ACL applied to the authenticated port in the inbound direction. |
| Outbound IPv6 ACL | The IPv6 ACL applied to the authenticated port in the inbound direction. |

## Examples

The following example displays 802.1X IP ACL authentication information for Ethernet interface 1/1/15.

```
device# show mac-authentication ip-acl ethernet 1/1/15
-----------------------------------------------------------------
Port      MAC              Inbound     Outbound   Inbound   Outbound
          Addr             IPv4 ACL    IPv4 ACL   IPv6 ACL  IPv6 ACL
-----------------------------------------------------------------
1/1/15     0180.c200.0003   10          11          20       21
1/1/15     0180.c300.0005   100         101         120      121
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | The output for this command was updated. |
| 08.0.70 | The command was modified to introduce the stack-id option. |

# show mac-authentication sessions

Displays MAC authentication sessions at the global and interface levels.

## Syntax

**show mac-authentication sessions** { **all** | **brief** | **stack-unit** *id* | **ethernet** *unit/slot/port* }

## Parameters

**all**
Displays MAC authentication sessions for all ports.

**brief**
Displays summary information for MAC authentication sessions.

**ethernet** *unit/slot/port*
Displays MAC sessions for the specified Ethernet interface.

**stack-unit** *id*
Displays MAC sessions for the specified stack unit.

## Modes

Privileged EXEC mode

## Usage Guidelines

A client session can have an IPv4 address and multiple IPv6 addresses. When multiple addresses exist, the **show mac-authentication sessions** command displays all addresses for the session.

## Command Output

The **show mac-authentication sessions** command displays the following information.

| Output field | Description |
|---|---|
| Port | Port number. |
| MAC Addr | MAC address of the client. |
| IP Addr | IP address or addresses of the client (a session can have an IPv4 address and multiple IPv6 addresses). IP addresses of the authenticated host are only displayed when an IP ACL is applied to the interface based on the RADIUS server response. |
| Vlan | VLAN ID. |
| Auth State | Authentication state. |
| ACL | Specific applied ACL. |
| Session Time | Session time. |
| Age | Age of the session. |

# Examples

The following example displays MAC authentication sessions for all interfaces.

```
device# show mac-authentication sessions all
--------------------------------------------------------------------------------
Port    MAC              IP(v4/v6)            VLAN  Auth      ACL    Session   Age
        Addr             Addr                       State           Time
--------------------------------------------------------------------------------
1/1/1   0024.38c9.da40   fe80::224:38ff:fec9: 100   Yes       None   7400      Ena
                         N/A
1/1/1   00aa.bbcc.dd00   fe80::2aa:bbff:fecc: 100   Yes       Yes    7400      Ena
                         222::223
                         100.100.100.10
```

The following example displays MAC authentication sessions for a specified interface.

```
device# show mac-authentication sessions ethernet 1/1/2
--------------------------------------------------------------------------------
Port        MAC              IP              Vlan  Auth      ACL    Session  Age
            Addr             Addr                 State           Time
--------------------------------------------------------------------------------
1/1/2       0010.94ab.0021   192.85.1.2      300   Yes       Yes    100      Ena
```

The following example displays MAC authentication sessions in brief.

```
device# show mac-authentication sessions brief
--------------------------------------------------------------------------------------------------
Port      Number of          Number of          Number of         Untagged          Dynamic
          Attempted Users    Authorized Users   Denied Users      VLAN Type         Port ACL
--------------------------------------------------------------------------------------------------
1/1/2     1                  1                  0                 Radius-VLAN         No
1/1/3     0                  0                  0                 Auth-Default-VLAN   No
1/1/4     0                  0                  0                 Auth-Default-VLAN   No
1/1/5     0                  0                  0                 Auth-Default-VLAN   No
2/1/1     0                  0                  0                 Auth-Default-VLAN   No
2/1/2     0                  0                  0                 Auth-Default-VLAN   No
2/1/4     0                  0                  0                 Auth-Default-VLAN   No
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | The command output was updated. |
| 08.0.61 | The command output was modified to display multiple IPv6 addresses for a session. |
| 08.0.70 | The command was modified to introduce the stack-unit option. |

# show mac-authentication sessions detail

Displays details of MAC authentication sessions for a specific interface.

## Syntax

**show mac-authentication sessions detail** { **ethernet** *unit/slot/port* }

## Parameters

**ethernet** *unit/slot/port*
Displays MAC sessions for the specified Ethernet interface.

## Modes

Privileged EXEC mode

## Usage Guidelines

A client session can have an IPv4 address and multiple IPv6 addresses. When multiple addresses exist, the **show mac-authentication sessions** command displays all addresses for the session.

## Command Output

The **show mac-authentication sessions detail** command displays the following information.

| Output field | Description |
|---|---|
| Port | Port number. |
| MAC Addr | MAC address of the client. |
| IP Addr | IP address or addresses of the client (a session can have an IPv4 address and multiple IPv6 addresses). IP addresses of the authenticated host are only displayed when an IP ACL is applied to the interface based on the RADIUS server response. |
| Vlan | VLAN ID. |
| Auth State | Authentication state. |
| ACL | Specific applied ACL. |
| Session Time | Session time. |
| Age | Age of the session. |

## Examples

The following example displays MAC authentication sessions for a specified interface.

```
device# show mac-authentication sessions ethernet 1/1/2
--------------------------------------------------------------------------------
Port        MAC             IP              Vlan  Auth      ACL    Session  Age
            Addr            Addr                  State           Time
--------------------------------------------------------------------------------
1/1/2       0010.94ab.0021  192.85.1.2      300   Yes       Yes    100      Ena
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show mac-authentication statistics

Displays the MAC authentication statistics.

## Syntax

**show mac-authentication statistics** { **all** | **stack-unit** *id* | **ethernet** *device**/**slot**/**port* }

## Parameters

**all**

Displays the MAC authentication statistics for all interfaces.

**ethernet** *device**/**slot**/**port*

Displays the MAC authentication statistics for the specified interface.

**stack-unit** *id*

Displays MAC-authentication statistics for the specified stack unit.

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show mac-authentication statistics** command displays the following information:

| Output field | Description |
|---|---|
| Accepted Sessions | The number of accepted sessions |
| Rejected Sessions | The number of rejected sessions |
| Inprogress Sessions | The number of in-progress sessions |
| Attempted Sessions | The number of attempted sessions |
| Number of Errors | The number of errors |

# Examples

The following example displays MAC authentication statistics for all interfaces.

```
device# show mac-authentication statistics all

Port 1/1/15 Statistics:
Accepted Sessions       :    2
Rejected Sessions       :    0
Inprogress Sessions     :    0
Attempted Sessions      :    0
Number of Errors        :    0

Port 2/1/15 Statistics:
Accepted Sessions       :    1
Rejected Sessions       :    0
Inprogress Sessions     :    0
Attempted Sessions      :    0
Number of Errors        :    0
```

The following example displays MAC authentication statistics for Ethernet interface 1/1/15.

```
device# show mac-auth statistics ethernet 1/1/15

Port 1/1/15 Statistics:
Accepted Sessions       :    2
Rejected Sessions       :    0
Inprogress Sessions     :    0
Attempted Sessions      :    0
Number of Errors        :    0
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.70 | The command was modified to introduce the stack-id option. |

# show macsec statistics

Displays status information and secure channel statistics for the designated MACsec interface.

## Syntax

**show macsec statistics ethernet** *device* **/** *slot* **/** *port*

## Parameters

*device***/***slot***/***port*

Interface for which MACsec status information is to be displayed. The interface is designated by device number in stack/slot on the device/interface on the slot.

**brief**

Specifies brief output for all MACsec interfaces.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

dot1x-mka configuration mode

dot1x-mka-interface configuration mode

## Usage Guidelines

MACsec commands are supported only on ICX 7450 and ICX 7650 devices.

It is recommended that you use the **clear macsec** command to clear previous results for the **show macsec statistics** command before re-executing it.

## Command Output

The **show macsec statistics** command displays the following information:

| Output field | Description |
|---|---|
| Interface (Device/slot/port) | The information that follows describes the designated interface. |
| Replay Protection (Enabled, Disabled) | Indicates whether replay protection is applied on the interface. |
| Replay Window (0 through 127) | If out-of-order packets are allowed, indicates allowable window within which an out-of-order packet can be received. |
| Frame Validation (Enabled, Disabled) | Indicates whether MACsec frame headers are checked. |
| Secure Channel Statistics: | The fields that follow describe activity on a secure channel established over the designated interface. |
| TxPktProtectedOnly | Number of transmitted packets with integrity protection only. |
| TxOctetProtectedOnly | Number of bytes transmitted in packets with integrity protection only. |

| Output field | Description |
|---|---|
| TxPktEncrypted | Number of transmitted packets that are encrypted. |
| TxOctetEncrypted | Number of bytes transmitted in encrypted packets. |
| TxPktMiss | Number of transmitted packets that are neither encrypted nor protected by integrity check. |
| TxOctetMiss | Number of bytes transmitted in packets that are neither encrypted nor protected by integrity checking. |
| TxPktDrop | Number of packets dropped at transmission because SAK has been exhausted. |
| TxPktBad | Number of transmitted packets marked as bad. |
| RxPktDecryptedAuth | Number of packets received, decrypted, and checked for integrity protection. |
| RxOctetTotal | Number of bytes received. |
| RxOctetAuthOnly | Number of bytes received with Integrity protection only. |
| RxOctetDecrypted | Number of bytes received and decrypted. |
| RxPktFailReplayCheck | Number of packets received out of order. |
| RxPktFailICVCheck | Number of packets received that failed Integrity checking. |
| RxPktNoMACsecTag | Number of packets received without a MACSec Tag. |
| RxPktFrameValFail | Number of packets received that failed MACsec frame validation. |
| RxPktMiss | Number of packets received that did not find a key for decryption. |
| RxOctetMiss | Number of bytes received that did not find a key for decryption. |
| RxPktDrop | Number of received packets that were dropped. |

# Examples

The following example shows output for an ICX 7450 device.

```
device(config)# clear macsec ethernet 10/2/1
device(config)# show macsec statistics ethernet 10/2/1
device(config)#
Interface Statistics:
---------------------
rx Untag Pkts             : 1                 tx Untag Pkts             : 0
rx Notag Pkts             : 0                 tx TooLong Pkts           : 0
rx Badtag Pkts            : 0
rx Unknownsci Pkts        : 0
rx Nosci Pkts             : 0
rx Overrun Pkts           : 0

Transmit Secure Channels:
-------------------------

SA[0] Statistics:
Protected Pkts            : 0
Encrypted Pkts            : 2436337

SA[1] Statistics:
Protected Pkts            : 0
Encrypted Pkts            : 0

SA[2] Statistics:
Protected Pkts            : 0
Encrypted Pkts            : 0

SA[3] Statistics:
Protected Pkts            : 0
Encrypted Pkts            : 0

SC Statistics:
Protected Octets          : 0                 Encrypted Octets          : 134830107
Protected Pkts            : 0                 Encrypted Pkts            : 2436337

Receive Secure Channels:
------------------------

SA[0] Statistics:
Ok Pkts                   : 1949642           Invalid Pkts              : 0
Not using SA Pkts         : 0                 Unused Pkts               : 0
Not Valid Pkts            : 0

SA[1] Statistics:
Ok Pkts                   : 0                 Invalid Pkts              : 0
Not using SA Pkts         : 0                 Unused Pkts               : 0
Not Valid Pkts            : 0

SA[2] Statistics:
Ok Pkts                   : 0                 Invalid Pkts              : 0
Not using SA Pkts         : 0                 Unused Pkts               : 0
Not Valid Pkts            : 0

SA[3] Statistics:
Ok Pkts                   : 0                 Invalid Pkts              : 0
Not using SA Pkts         : 0                 Unused Pkts               : 0
Not Valid Pkts            : 0

SC Statistics:
OkPkts                    : 1949642           Invalid Pkts              : 0
Not using SA Pkts         : 0                 Unused Pkts               : 0
Not Valid Pkts            : 0                 Unchecked Pkts            : 0
Delayed Pkts              : 0                 Late Pkts                 : 0
Valid Octets              : 0                 Decrypted Octets          : 97743896
device(config)#
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.20a | This command was modified. The **show macsec** command name was changed to **show macsec statistics** . |
| 08.0.30 | Support for this command was added on ICX 7450 devices. |
| 08.0.70 | Support for this command was added on ICX 7650 devices. |

# show management traffic exclusion

Displays the port types and application types that are excluded from in-band or out-of-band (OOB) management ports.

## Syntax

**show management traffic exclusion**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

## Examples

To display port and application status:

```
device# show management traffic exclusion
Port        App
Inband      all
oob         all
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show management-vrf

Displays management Virtual Routing and Forwarding (VRF) packet and session rejection statistics including dropped packets due to failure in management VRF validation.

## Syntax

**show management-vrf**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

Ensure that the management VRF is configured before executing the **show management-vrf** command.

## Command Output

The **show management-vrf** command displays the following information.

| Output field | Description |
|---|---|
| Management VRF name | Displays the configured management VRF name. |
| Management Application | Displays the management application names. |
| Rx Drop Pkts | Displays the number of packets dropped in the inbound traffic. |
| Tx Drop Pkts | Displays the number of packets dropped in the outbound traffic. |
| TCP Connection rejects | Displays the number of TCP connections per application rejected due to management VRF validation. |

## Examples

The following is sample output from the **show management-vrf** command.

```
device(config)# show management-vrf

Management VRF name : sflow
Management Application     Rx Drop Pkts     Tx Drop Pkts
SNMP Engine              0                  11
RADIUS Client              0               0
TFTP Client              0                 0
Traps                      -               0
SysLogs                  -                 0
TCP Connection rejects:
Telnet              : 0
SSH (Strict)      : 685
TACACS+ Client    : 0
```

# show spx pe-port-vlan-resources

Displays information on maximum VLANs-per-PE port configuration in a Campus Fabric (SPX) system.

## Syntax

**show spx pe-port-vlan-resources**

## Modes

Privileged Exec mode or any configuration mode.

## Command Output

The **show spx pe-port-vlan-resources** command displays the following information:

| Output field | Description |
|---|---|
| Reserved number of VLANs for each PE port | 4 (fixed value). Number of VLANs for which each PE port is guaranteed membership. |
| Maximum number of VLANs a PE port can be added to | Maximum allowable VLANs per PE port as configured with the max-vlans-per-pe-port (Valid range is 5 through 1024). |
| PE VLAN Global Pool size | 4096 (fixed value). Global pool size for PE port-to-VLAN assignments, not including four VLAN assignments already reserved for each PE port by default. |
| Entries used in PE VLAN Global Pool | Number of configured entries in the global PE VLAN pool. The number may be oversubscribed. |
| Entries available in PE VLAN Global Pool | From 4096, number of remaining entries in the PE VLAN pool. |
| Ports with more than reserved number of VLANs | Number of PE ports that have more than 4 VLANs configured. |
| Configuration failures due to hash collisions | Number of PE port VLAN configuration rejections due to table space collision at or near maximum scaling. |

# Examples

The following sample output of the **show spx pe-port-vlan-resources** command displays the default VLAN-per-PE configuration. The PE VLAN global pool is empty.

```
ICX7750-48F Router# show spx pe-port-vlan-resources

PE VLAN Global Pool Resource Usage:

  ============================================================
  Reserved number of VLANs for each PE port              : 4
  Maximum number of VLANs a PE port can be added to      : 1024
  PE VLAN Global Pool size                               : 4096
  Entries used in PE VLAN Global Pool                    : 0
  Entries available in PE VLAN Global Pool               : 4096
  Ports with more than reserved number of VLANs          : 0
  Configuration failures due to hash collisions          : 0
  ============================================================

ICX7750-48F Router# show spx pe-port-vlan-resources

PE VLAN Global Pool Resource Usage:

  ============================================================
  Reserved number of VLANs for each PE port              : 4
  Maximum number of VLANs a PE port can be added to      : 1024
  PE VLAN Global Pool size                               : 4096
  Entries used in PE VLAN Global Pool                    : 0
  Entries available in PE VLAN Global Pool               : 4096
  Ports with more than reserved number of VLANs          : 0
  Configuration failures due to hash collisions          : 0
  ============================================================
```

The following example shows a system with a depleted PE VLAN pool.

```
ICX7750-48F Router(config-spx-cb)# show spx pe-port-vlan-resources
PE VLAN Global Pool Resource Usage:
============================================================
Reserved number of VLANs for each PE port              : 4
Maximum number of VLANs a PE port can be added to      : 1023
Configured PE VLAN entries                             : 4483
PE VLAN Global Pool size                               : 4096
Entries used in PE VLAN Global Pool                    : 4096
Entries available in PE VLAN Global Pool               : 0
Ports with more than reserved number of VLANs          : 33
Configuration failures due to hash collisions          : 0
============================================================
```

# History

| Release version | Command history |
|---|---|
| Release 08.0.80 | This command was introduced. |

# show media

Displays information about the media devices installed per device, per stack, and per port.

## Syntax

**show media** [ **validation** ] [ **ethernet** *unit/slot/port* | **stack** *stack* ]

## Parameters

**validation**
Displays detailed information about the optics inventory and shows if the optics are official Ruckus optics.

**ethernet** *unit/slot/port*
Displays the media type for the specified Ethernet interface.

**stack** *stack*
Displays the media type for the specified stack.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

802.1br PE local mode

## Command Output

The show flash command displays the Type, Vendor, Part number, Version and Serial number of the SFP, SFP+, QSPF, or QSPF+ optical device installed in the port. If none of these optical devices are installed in a port, the "Type" field will display "EMPTY".

# Examples

The following example sample output displays information about the media installed on a device.

```
device# show media

Port 1/1/1 :     Type : 1G M-C (Gig-Copper)
Port 1/1/2 :     Type : 1G M-C (Gig-Copper)
Port 1/1/3 :     Type : 1G M-C (Gig-Copper)
Port 1/1/4 :     Type : 1G M-C (Gig-Copper)
Port 1/1/5 :     Type : 1G M-C (Gig-Copper)
Port 1/1/6 :     Type : 1G M-C (Gig-Copper)
Port 1/1/7 :     Type : 1G M-C (Gig-Copper)
Port 1/1/8 :     Type : 1G M-C (Gig-Copper)
Port 1/1/9 :     Type : 1G M-C (Gig-Copper)
Port 1/1/10:     Type : 1G M-C (Gig-Copper)
Port 1/1/11:     Type : 1G M-C (Gig-Copper)
Port 1/1/12:     Type : 1G M-C (Gig-Copper)
Port 1/1/13:     Type : 1G M-C (Gig-Copper)
Port 1/1/14:     Type : 1G M-C (Gig-Copper)
Port 1/1/15:     Type : 1G M-C (Gig-Copper)
Port 1/1/16:     Type : 1G M-C (Gig-Copper)
Port 1/1/17:     Type : 1G M-C (Gig-Copper)
Port 1/1/18:     Type : 1G M-C (Gig-Copper)
Port 1/1/19:     Type : 1G M-C (Gig-Copper)
Port 1/1/20:     Type : 1G M-C (Gig-Copper)
Port 1/1/21:     Type : 1G M-C (Gig-Copper)
Port 1/1/22:     Type : 1G M-C (Gig-Copper)
Port 1/1/23:     Type : 1G M-C (Gig-Copper)
Port 1/1/24:     Type : 1G M-C (Gig-Copper)
Port 1/2/1 :     Type : 10GE SR 300m (SFP +)
Port 1/2/2 :     Type : EMPTY
Port 1/2/3 :     Type : 1G Twinax 1m (SFP)
Port 1/2/4 :     Type : 1G Twinax 1m (SFP)
```

Use the **show media validation** command to detail the optics inventory and show if these optics are official Ruckus optics.

**NOTE**
Ruckus supports digital optical monitoring only on Ruckus optics.

```
device# show media validation
Port       Supported Vendor
Type
--------------------------------------------------------------------------
1/1/1      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/2      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/3      Yes       RUCKUS               Type  : 10GE USR 100m (SFP
+)
1/1/4      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/5      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/6      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/7      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/8      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/9      Yes       RUCKUS               Type  : 1G M-
GBXD(SFP)
1/1/11     Yes       RUCKUS               Type  : 1G M-
GBXU(SFP)
1/1/13     Yes       RUCKUS               Type  : 1G M-
LHA(SFP)
1/1/14     Yes       RUCKUS               Type  : 1G M-
LHA(SFP)
1/1/17     Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
1/1/19     Yes       RUCKUS               Type  : 10GE USR 100m (SFP
+)
1/1/24     Yes       RUCKUS               Type  : 1GE M-
SX(SFP)
1/1/25     Yes       RUCKUS               Type  : 1G M-
TX(SFP)
1/1/26     Yes       RUCKUS               Type  : 1G M-
TX(SFP)
1/1/27     Yes       RUCKUS               Type  : 1G M-
TX(SFP)
1/1/28     Yes       RUCKUS               Type  : 1G M-
TX(SFP)
1/1/30     Yes       RUCKUS               Type  : 1G M-
TX(SFP)
1/1/31     Yes       RUCKUS               Type  : 1G M-
GBXU(SFP)
1/1/33     Yes       RUCKUS               Type  : 1G M-
GBXD(SFP)
1/1/37     Yes       RUCKUS               Type  : 1G M-
GBXD(SFP)
1/1/39     Yes       RUCKUS               Type  : 1G M-
TX(SFP)
1/1/43     Yes       RUCKUS               Type  : 1G M-
GBXU(SFP)
1/1/47     Yes       RUCKUS               Type  : 10GE USR 100m (SFP
+)
1/1/48     No        FINISAR CORP.        Type  : 1GE M-
SX(SFP)
1/2/1      Yes       RUCKUS               Type  : 40GE-SR4 100m (QSFP
+)
1/3/1      Yes       RUCKUS               Type  : 40GE-Active Copper 1m (QSFP
+)
1/3/4      Yes       RUCKUS               Type  : 40GE-Active Copper 1m (QSFP
+)
```

```
2/1/1      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
2/1/2      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
2/1/3      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
2/1/4      Yes       RUCKUS               Type  : 10GE SR 300m ((SFP
+))
2/1/5      Yes       RUCKUS               Type  : 1G M-
TX(SFP)
2/1/6      Yes       RUCKUS               Type  : 1G M-
TX(SFP)
2/1/9      Yes       RUCKUS               Type  : 1GE M-SX(SFP)
```

The following sample output displays the media type for a specified stack.

```
device# show media stack 1
Port 1/1/1   :  Type  : EMPTY
Port 1/1/2   :  Type  : EMPTY
Port 1/1/3   :  Type  : EMPTY
Port 1/1/4   :  Type  : EMPTY
Port 1/1/5   :  Type  : EMPTY
Port 1/1/6   :  Type  : EMPTY
Port 1/1/7   :  Type  : EMPTY
Port 1/1/8   :  Type  : EMPTY
Port 1/1/9:1 :  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/9:2 :  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/9:3 :  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/9:4 :  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/10  :  Type  : EMPTY
Port 1/1/11:1:  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/11:2:  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/11:3:  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/11:4:  Type  : 4x10GE Active Copper 1m (QSFP+)
Port 1/1/12  :  Type  : EMPTY
Port 1/1/13  :  Type  : EMPTY
Port 1/1/14  :  Type  : EMPTY
Port 1/1/15  :  Type  : EMPTY
Port 1/1/16  :  Type  : EMPTY
Port 1/1/17  :  Type  : EMPTY
Port 1/1/18  :  Type  : EMPTY
Port 1/1/19  :  Type  : EMPTY
Port 1/1/20  :  Type  : EMPTY
Port 1/2/1   :  Type  : EMPTY
Port 1/2/2   :  Type  : EMPTY
Port 1/2/3   :  Type  : EMPTY
Port 1/2/4   :  Type  : EMPTY
Port 1/2/5   :  Type  : EMPTY
Port 1/2/6   :  Type  : EMPTY
Port 1/3/1   :  Type  : 40GE-SR4 100m (QSFP+)
Port 1/3/2   :  Type  : EMPTY
Port 1/3/3   :  Type  : EMPTY
Port 1/3/4   :  Type  : EMPTY
Port 1/3/5   :  Type  : EMPTY
Port 1/3/6   :  Type  : EMPTY
Port 17/1/2  :  Type  : 1G M-C (Gig-Copper)
Port 17/1/3  :  Type  : 1G M-C (Gig-Copper)
Port 17/1/4  :  Type  : 1G M-C (Gig-Copper)
Port 17/1/5  :  Type  : 1G M-C (Gig-Copper)
Port 17/1/6  :  Type  : 1G M-C (Gig-Copper)
Port 17/1/7  :  Type  : 1G M-C (Gig-Copper)
...
```

The following sample output displays the media type for the specified Ethernet interface.

```
device# show media ethernet 1/3/1
Port    1/3/1: Type  : 40GE-SR4 100m (QSFP+)
Vendor: RUCKUS            Version: A
Part# : 57-1000128-01     Serial#: LTA112251000543
```

The folllowing sample output displays the media type for an 802.1br CB interface from the local display of an attached PE unit.

```
device# show media ethernet 4/4/1
Port   4/4/1: Type  : 40GE-LR4  10km (QSFP+ LC)
            Vendor: RUCKUS            Version: 2
            Part# : 57-1000263-01     Serial#: LDJ21325C230002
```

# show memory

Displays the memory usage for system tasks, transmission control protocol, and stack units.

## Syntax

**show memory** [ **task** | **tcp** | **unit** *unit-id* ]

## Parameters

**task**

Displays memory usage per system task.

**tcp**

Displays Transmission Control Protocol (TCP) memory usage.

**unit** *unit-id*

The ID of the stack unit.

## Modes

Global configuration mode

User EXEC mode

## Usage Guidelines

## Command Output

The **show memory task** command displays the following information:

| Output field | Description |
|---|---|
| Task | The name of the task. |
| Alloc | The amount memory allocated for the task. |
| Free | The amount of free memory available. |
| Used | The amount of memory used by the specific task. |
| TCB usage | The availability of Transmission Control Block for the TCP connection. |
| TCP QUEUE BUFFER usage | The availability of the Queue buffer used to hold the TCP messages that need to be sent. |
| TCP SEND BUFFER usage | The availability of buffers which will be used to send the TCP packets from the device. |
| TCP RECEIVE BUFFER usage | The availability of buffers which will be used to receive the TCP packets to the device. |
| TCP OUT OF SEQUENCE BUFFER usage | The availability of re-sequence buffer used for the TCP connection. |

# Examples

The following example command displays the memory usage per task.

```
Task Memory Usage Info
------------------------------
Last clear : NA
----------------------------------------------------------------------
Task                          Alloc         Free          Used
----------------------------------------------------------------------
TimerTsk                        144            0           144
FlashTsk                       5552            0          5552
MainTsk                    33153780      3411177      29742603
keygen                          1468            0          1468
itc                             9188            0          9188
bcmCNTR.0                      17820            0         17820
bcmL2MOD.0                       144            0           144
scp                            232815        27166        205649
appl                        676257682    637313495      38944187
snms                           127713        52104         75609
rtm                           9476869        17272       9459597
rtm6                           321341        17272        304069
rip                            574422         8636        565786
bgp                           4048555        17272       4031283
ospf                          2937465         8636       2928829
openflow_ofm                   431242        14621        416621
openflow_opm                   433909        17272        416637
mcast_fwd                     1776859        17272       1759587
mcast                         2614790        31233       2583557
msdp                           221375        17272        204103
ripng                           96181         8636         87545
ospf6                         1989857         8636       1981221
mcast6                         794175        22597        771578
ipsec                          208381         8636        199745
dhcp6                          134907         8636        126271
snmp                            57140        17272         39868
rmon                            74775        17272         57503
web                             56915        17344         39571
acl                           1291591        28243       1263348
flexauth                       277607         8636        268971
ntp                             56835        17272         39563
rconsole                        48215         8636         39579
console                       2059410      1476779        582631
ospf_msg_task                   56035        17272         38763
auxTsk                           4572            0          4572
bcmLINK.0                       37152        37152             0
Total Memory Used: 97213162
```

The following example displays the TCP memory usage information.

```
device# show memory tcp
TCP MEMORY USAGE
TCB usage: total=73140, free=71300
TCP QUEUE BUFFER usage: total=19635, free=19635
TCP SEND BUFFER usage: total=192532, free=192532
TCP RECEIVE BUFFER usage: total=192532, free=192532
TCP OUT OF SEQUENCE BUFFER usage: total=25074, free=25074
```

The following example displays memory usage for stack unit 1.

```
device# show memory unit 1
Stack unit 1:
  Total DRAM: 268435456 bytes
  Dynamic memory: 3781353472 bytes total, 3563307008 bytes free, 5% used
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show memory task

Displays the memory usage, allocated memory, and free memory for system tasks on the device.

## Syntax

**show memory task** [ **clear** ]

## Parameters

**clear**

Clears the displayed memory information if no memory is used.

## Modes

Global configuration mode

User EXEC mode

# Usage Guidelines

# Examples

The following example displays the memory usage, allocated memory, and free memory for system tasks on the device.

```
device# show memory task
Task Memory Usage Info
----------------------
Last clear : NA
----------------------------------------------------------------------
 Task                          Alloc         Free        Used
----------------------------------------------------------------------
TimerTsk                         144            0         144
FlashTsk                        5552            0        5552
MainTsk                     33153780      3411177    29742603
keygen                         1468            0        1468
itc                            9188            0        9188
bcmCNTR.0                     17820            0       17820
bcmL2MOD.0                      144            0         144
scp                          232815        27166      205649
appl                      676257682    637313495    38944187
snms                         127713        52104       75609
rtm                         9476869        17272     9459597
rtm6                         321341        17272      304069
rip                          574422         8636      565786
bgp                         4048555        17272     4031283
ospf                        2937465         8636     2928829
openflow_ofm                 431242        14621      416621
openflow_opm                 433909        17272      416637
mcast_fwd                   1776859        17272     1759587
mcast                       2614790        31233     2583557
msdp                         221375        17272      204103
ripng                         96181         8636       87545
ospf6                       1989857         8636     1981221
mcast6                       794175        22597      771578
ipsec                        208381         8636      199745
dhcp6                        134907         8636      126271
snmp                          57140        17272       39868
rmon                          74775        17272       57503
web                           56915        17344       39571
acl                         1291591        28243     1263348
flexauth                     277607         8636      268971
ntp                           56835        17272       39563
rconsole                      48215         8636       39579
console                     2059410      1476779      582631
ospf_msg_task                 56035        17272       38763
auxTsk                         4572            0        4572
bcmLINK.0                     37152        37152           0
Total Memory Used: 97213162
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show metro-ring

Displays the metro ring details.

## Syntax

**show metro-ring** *ring-id* [ **diagnostics** ]

## Parameters

*ring-id*

Displays the details of the metro ring specified by the ring ID.

**diagnostics**

Displays the diagnostic results for the specified metro ring.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

VSRP VRID configuration mode

## Command Output

The **show metro-ring** *ring-id* **diagnostics** command displays the following information:

| Output field | Description |
|---|---|
| Ring id | The metro ring ID. |
| Diag state | The state of ring diagnostics. |
| RHP average time | The average round-trip time for an Ring Hello Packet (RHP) packet on the ring. The calculated time has a granularity of 1 microsecond. |
| Recommended hello time | The hello time recommended by the software based on the RHP average round-trip time. |
| Recommended Prefwing time | The preforwarding time recommended by the software based on the RHP average round-trip time. |
| Diag frame sent | The number of diagnostic RHPs sent for the test. |
| Diag frame lost | The number of diagnostic RHPs lost during the test. |

The **show metro-ring** *ring-id* command displays the following information:

| Output field | Description |
|---|---|
| Ring id | The metro ring ID. |
| State | The state of MRP. The state can be enabled or disabled. |

| Output field | Description |
|---|---|
| Ring role | Whether this node is the master for the ring. The role can be master or member. |
| Master vlan | The ID of the master VLAN in the topology group used by this ring. If a topology group is used by MRP, the master VLAN controls the MRP settings for all VLANs in the topology group. The topology group ID is 0 if the MRP VLAN is not the master VLAN in a topology group. Using a topology group for MRP configuration is optional. |
| Topo group | The topology group ID. |
| Hello time | The interval, in milliseconds, at which the forwarding port on the ring master node sends RHPs. |
| Prefwing time | The number of milliseconds an MRP interface that has entered the preforwarding state will wait before changing to the forwarding state. |
| Ring interfaces | The ring interfaces in the device. |
| Interface role | The interface role can be one of the following:<br>• primary<br>– Master node - The interface generates RHPs.<br>– Member node - The interface forwards RHPs received on the other interface (the secondary interface).<br>• secondary - The interface does not generate RHPs.<br>– Master node - The interface listens for RHPs.<br>– Member node - The interface receives RHPs. |
| Forwarding state | Whether MRP forwarding is enabled on the interface. The forwarding state can be one of the following:<br>• blocking - The interface is blocking Layer 2 data traffic and RHPs.<br>• disabled - The interface is down.<br>• forwarding - The interface is forwarding Layer 2 data traffic and RHPs.<br>• preforwarding - The interface is listening for RHPs but is blocking Layer 2 data traffic. |
| Active interface | The physical interfaces that are sending and receiving RHPs. If a port is disabled, its state is shown as "disabled". If an interface is part of a LAG, the member port which comes up first is listed. |
| Interface Type | Shows if the interface is a regular port or a tunnel port. |
| RHPs sent | The number of RHPs sent on the interface.<br>**NOTE**<br>This field applies only to the master node. On non-master nodes, this field contains 0. This is because the RHPs are forwarded in hardware on the non-master nodes. |
| RHPs rcvd | The number of RHPs received on the interface.<br>**NOTE**<br>On most devices, this field applies only to the master node. On non-master nodes, this field contains 0. This is because the RHPs are forwarded in hardware on the non-master nodes. However, on the FastIron devices, the RHP received counter on non-master MRP nodes increments. This is because, on FastIron devices, the CPU receives a copy of the RHPs forwarded in hardware. |
| TC RHPs rcvd | The number of Topology Change RHPs received on the interface. A Topology Change RHP indicates that the ring topology has changed. |
| State changes | The number of MRP interface state changes that have occurred. The state can be one of the states listed in the Forwarding state field. |

# Examples

The following example displays the MRP diagnostics result on the master node.

```
device# show metro-ring 1 diagnostics
Metro Ring 1 - custA
=============
diagnostics results

Ring       Diag       RHP average     Recommended     Recommended
id         state      time(microsec)  hello time(ms)  Prefwing time(ms)
1          disabled   < 0             100             300

Diag frame sent      Diag frame lost
0                    0
```

The following example displays the output of the **show metro-ring** command.

```
device# show metro-ring 1
Metro Ring 1
=============
Ring     State     Ring     Master    Topo       Hello      Prefwing
id                 role     vlan      group      time(ms)   time(ms)
2        enabled   member   2         not conf   100        300
Ring interfaces    Interface role     Forwarding state   Active interface  Interface Type
ethernet 1/1/1     primary            disabled           none              Regular
ethernet 1/1/2     secondary          forwarding         ethernet 2        Tunnel
RHPs sent          RHPs rcvd       TC RHPs rcvd       State changes
3                  0                                                0                 4
```

# show mirror

Displays the port mirroring configuration details.

## Syntax

**show mirror ethernet** *stackid/slot/port*

## Parameters

**ethernet** *stackid/slot/port*

Displays the details for the specified Ethernet port.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Examples

The following example displays sample output of the **show mirror** command.

```
device(config)# show mirror ethernet 1/2/1
Mirror port 1/2/1
  Input monitoring      : (U1/M1)   2
  Output monitoring     : None
```

# show module

Displays module information for stack members.

## Syntax

**show module**

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

## Command Output

The **show module** command displays the following information:

| Output field | Description |
|---|---|
| Module | Identifies the module by stack unit ID, module number, and module type. |
| Status | The status of this module. |
| Ports | The number of ports in this module. |
| Starting MAC | The starting MAC address for this module. |

# Examples

The following example displays stack module information.

```
device# show module

       Module                                          Status Ports Starting MAC
U1:M1  ICX7450-48 48-port Management Module              OK     48   cc4e.248d.f8d0
U1:M2  ICX7400-4X10GF 4-port 40G Module                 OK     4    cc4e.248d.f901
U1:M3  ICX7400-1X40GQ 1-port 40G Module                 OK     1    cc4e.248d.f905
U2:M1  ICX7450-48 48-port Management Module              OK     48   cc4e.248e.4990
U2:M2  ICX7400-4X10GF 4-port 40G Module                 OK     4    cc4e.248e.49c1
U2:M3  ICX7400-SERVICE-MOD Module                       OK     0
U2:M4  ICX7400-1X40GQ 1-port 40G Module                 OK     1    cc4e.248e.49c9
U3:M1  ICX7450-48 48-port Management Module              OK     48   cc4e.248e.4490
U3:M2  ICX7400-4X10GF 4-port 40G Module                 OK     4    cc4e.248e.44c1
U3:M3  ICX7400-1X40GQ 1-port 40G Module                 OK     1    cc4e.248e.44c5
U3:M4  ICX7400-1X40GQ 1-port 40G Module                 OK     1    cc4e.248e.44c9
```

The following example displays stack module information when a module is removed form the device.

```
device#show module
        Module                                                Status Ports Starting MAC
U1:M1  ICX7450-24P POE 24-port Management Module       OK     24   cc4e.248e.5648
U1:M2  ICX7400-4X10GF 4-port 40G Module                CFG     4   cc4e.248e.5665
U1:M3  ICX7400-1X40GQ 1-port 40G Module                OK      1   cc4e.248e.5665
U1:M4  ICX7400-1X40GQ 1-port 40G Module                OK      1   cc4e.248e.5669
device#
```

# show monitor

Displays the monitored ports configurations.

## Syntax

**show monitor ethernet** *stackid/slot/port*

## Parameters

**ethernet** *stackid/slot/port*

Displays the information for the specified monitored Ethernet port.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example displays sample output of the **show monitor** command.

```
device> show monitor ethernet 1/1/2
  Input mirrored by     : (U1/M2)   1
  Output mirrored by    : None
```

# show mstp

Displays the MSTP information.

## Syntax

**show mstp** { [ **detail** ] *mstp-id* | **configuration** }

## Parameters

**detail**

Displays detailed MSTP information for the specified ID.

*mstp-id*

Displays the MSTP information for a specific ID.

**configuration**

Displays MSTP configuration information.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

## Command Output

The **show mstp** command displays the following information:

| Output field | Description |
|---|---|
| MSTP Instance | The ID of the MSTP instance whose statistics are being displayed. For the CIST, this number is 0. |
| VLANs | The number of VLANs that are included in this instance of MSTP. For the CIST, this number will always be 1. |
| Bridge Identifier | The MAC address of the bridge. |
| Bridge MaxAge sec | Displays the configured maximum age. |
| Bridge Hello sec | Displays the configured Hello variable. |
| Bridge FwdDly sec | Displays the configured FwdDly variable. |
| Bridge Hop cnt | Displays the configured Max Hop count variable. |
| Root MaxAge sec | The maximum age configured on the root bridge. |
| Root Hello sec | Hello interval configured on the root bridge. |
| Root FwdDly sec | FwdDly interval configured on the root bridge. |
| Root Hop Cnt | Maximum hop count left from the root bridge. |
| Root Bridge | Bridge identifier of the root bridge. |

| Output field | Description |
|---|---|
| ExtPath Cost | The configured path cost on a link connected to this port to an external MSTP region. |
| Regional Root Bridge | The Regional Root Bridge is the MAC address of the root bridge for the local region. |
| IntPath Cost | The configured path cost on a link connected to this port within the internal MSTP region. |
| Designated Bridge | The MAC address of the bridge that sent the best BPDU that was received on this port. |
| Root Port | Port indicating shortest path to root. Set to "Root" if this bridge is the root bridge. |
| Port Num | The port number of the interface. |
| Pri | The configured priority of the port. The default is 128. |
| PortPath Cost | Configured or auto-detected path cost for port. |
| P2P Mac | Indicates if the port is configured with a point-to-point link:<br>• T - The port is configured in a point-to-point link.<br>• F - The port is not configured in a point-to-point link. |
| Edge | Indicates if the port is configured as an operational edge port:<br>• T - Indicates that the port is defined as an edge port.<br>• F - Indicates that the port is not defined as an edge port. |
| Role | The port current spanning tree state. A port can have one of the following states:<br>• Forwarding<br>• Discarding<br>• Learning<br>• Disabled |
| Designated Cost | Port path cost to the root bridge. |
| Max Hop cnt | The maximum hop count configured for this instance. |

# Examples

The following example displays the MSTP information for a specified MSTP instance.

```
device# show mstp 1
MSTP Instance 1 - VLANs: 2
-------------------------------------------------------------------------------
Bridge            Max          RegionalRoot       IntPath     Designated       Root    Root
Identifier        Hop          Bridge             Cost        Bridge           Port    Hop
hex               cnt          hex                            hex                      cnt
8001000cdb80af01  20           8001000cdb80af01   0           8001000cdb80af01  Root   20
Port    Pri       PortPath     Role State         Designa-    Designated
Num     Cost                                     ted cost    bridge
3/1 128 2000      MASTER       FORWARDING         0           8001000cdb80af01
```

The following example displays the detailed MSTP information.

```
device# show mstp detail
MSTP Instance 0 (CIST) - VLANs: 4093
-------------------------------------------------------------------------------
Bridge: 800000b000c00000 [Priority 32768, SysId 0, Mac 00b000c00000]
FwdDelay 15, HelloTime 2, MaxHops 20, TxHoldCount 6
Port 6/54 - Role: DESIGNATED - State: FORWARDING
PathCost 20000, Priority 128, OperEdge T, OperPt2PtMac F, Boundary T
Designated - Root 800000b000c00000, RegionalRoot 800000b000c00000,
Bridge 800000b000c00000, ExtCost 0, IntCost 0
ActiveTimers - helloWhen 1
MachineState - PRX-DISCARD, PTX-IDLE, PPM-SENDING_RSTP, PIM-CURRENT
PRT-ACTIVE_PORT, PST-FORWARDING, TCM-INACTIVE
BPDUs - Rcvd MST 0, RST 0, Config 0, TCN 0
Sent MST 6, RST 0, Config 0, TCN 0
```

The following example displays the MSTP configuration details.

```
device# show mstp configuration
MSTP CONFIGURATION
------------------
Name : Reg1
Revision : 1
Version : 3 (MSTP mode)
Status : Started
Instanc  VLANs
-------- --------------
0        4093
```

# show mstp root-protect

Displays the MSTP root-protect information.

## Syntax

**show mstp root-protect**

## Modes

Global configuration mode

## Examples

To verify whether MSTP instances are in consistent state or in Inconsistent state, enter the following command that displays the MSTP root-protect information.

```
device# show mstp root-protect
Port    MSTI     Current State
1/1/5    MSTI 1 Consistent state

1/1/5   CIST     Inconsistent state (59 seconds left on timer)
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show notification mac-movement

Displays the MAC address movement notifications.

## Syntax

**show notification mac-movement** { **interval-history** | **threshold-rate** }

## Parameters

**interval-history**
Displays the collected history of MAC address movement notification and how the history interval is configured.

**threshold-rate**
Displays the configuration of the MAC address movement threshold rate.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show notification mac-movement interval-history** command displays the following information:

| Output field | Description |
|---|---|
| Interval-History Mac Movement Notification | Specifies whether the interval history data collection is enabled. |
| Configured Interval | The interval over which the MAC address movement statistics were collected. |
| Number of macs that moved in the interval | The number of MAC addresses that moved during the configured interval regardless of how many times each address moved. |
| Total number of moves in the interval | The total number of MAC address moves over the configured interval. |
| Interval Move-Count | The number of times the MAC address has moved within the interval. |

The **show notification mac-movement threshold-rate** command displays the following information:

| Output field | Description |
|---|---|
| Threshold-Rate Mac Movement Notification | Specifies whether the MAC movement notification threshold rate is enabled. |
| Configured Threshold-Rate | The rate in MAC address moves per sampling interval after which a notification is issued. The range is from 1 through 50000. |
| Configured Sampling-Interval | The sampling interval in seconds over which the number of MAC address moves is measured. The range is from 1 through 86400, which is the number of seconds in a day. |

| Output field | Description |
|---|---|
| Number of entries in the notification table | One entry for each time a MAC address notification threshold was reached. |
| MAC-Address | The MAC address that has moved to a different port. |
| from-Port | The port from which the MAC address moved. |
| to-Port | The port to which the MAC address moved. |
| Last Move-Time | The time the last move occurred. The system uptime is used if there is no time server configured. |
| Vlan-id | The VLAN for the port where the MAC address movement was detected. |

# Examples

The following example displays the notification interval history.

```
device# show notification mac-movement interval-history
Interval-History Mac Movement Notification is ENABLED
Configured Interval : 30 seconds
Number of macs that moved in the interval : 100
Total number of moves in the interval : 98654
MAC-Address       from-Port   to-Port   Interval Move-Count   Last Move-Time   Vlan-id
--------------    ---------   -------   -------------------   --------------   -------

0000.0000.0052    1/7/1      1/7/2        1000                 May 15 01:13:20   10
0000.0000.0051    1/7/1      1/7/2        1002                 May 15 01:13:20   10
0000.0000.0050    1/7/1      1/7/2        1012                 May 15 01:13:20   10
0000.0000.004f    1/7/1      1/7/2        1018                 May 15 01:13:20   10
0000.0000.004e    1/7/1      1/7/2        1012                 May 15 01:13:20   10
(output truncated)
```

The following examples displays the notification for a threshold rate.

```
device# show notification mac-movement threshold-rate
Threshold-Rate Mac Movement Notification is ENABLED
Configured Threshold-Rate : 5 moves
Configured Sampling-Interval : 30 seconds
Number of entries in the notification table : 100
MAC-Address       from-Port    to-Port    Last Move-Time   Vlan-id
--------------    ---------    -------    --------------   -------
0000.0000.0022    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.0021    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.0020    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.001f    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.0024    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.001e    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.0023    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.001d    1/7/1        1/7/2        Apr 29 18:29:35  10
0000.0000.001c    1/7/1        1/7/2        Apr 29 18:29:35 10
(output truncated)
```

# show notification-mac

Displays whether MAC-notification for SNMP traps is enabled or disabled.

## Syntax

**show notification-mac**

## Modes

Privileged EXEC mode

## Usage Guidelines

You can view statistics such as the configured interval, the number of traps sent, and the number of events sent.

## Examples

The following example displays the MAC-notification statistics:

```
device# show notification-mac
Mac-notification SNMP trap is ENABLED
Configured Interval: 40 seconds
Number of trap messages sent: 2
Number of mac-notification events sent: 20
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# show ntp associations

Displays association information for all NTP servers and peers.

## Syntax

**show ntp associations** [ **detail** [ *ipv4-address* | *ipv6-address* ] ]

## Parameters

**detail**

Displays the detailed NTP server and peer association information for the specifies address.

*ipv4-address*

Displays the NTP server and peer association information for a specific IPv4 address.

*ipv6-address*

Displays the NTP server and peer association information for a specific IPv6 address.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

NTP configuration mode

Interface configuration mode

## Command Output

The **show ntp associations** command displays the following information.

| Output Field | Description |
|---|---|
| * | The peer has been declared the system peer and lends its variables to the system variables. |
| # | This peer is a survivor in the selection algorithm. |
| + | This peer is a candidate in the combine algorithm. |
| - | This peer is discarded as an outlier in the clustering algorithm. |
| x | This peer is discarded as a "falseticker" in the selection algorithm. |
| ~ | The server or peer is statically configured. |
| address | IPv4 or IPv6 address of the peer. |
| ref clock | IPv4 address or first 32 bits of the MD5 hash of the IPv6 address of the peer to which the clock is synchronized. |
| st | Stratum setting for the peer. |
| when | Time, in seconds, since the last NTP packet was received from the peer. |
| poll | Polling interval (seconds). |
| reach | Peer reachability (bit string, in octal). |

| Output Field | Description |
|---|---|
| delay | Round-trip delay to the peer, in milliseconds. |
| offset | Relative time difference between a peer clock and a local clock, in milliseconds. |
| disp | Dispersion. |

The **show ntp associations detail** command displays the following information.

| Output field | Description |
|---|---|
| server | Indicates that the server is statically configured. |
| symmetric active peer | Indicates that the peer is statically configured. |
| symmetric passive peer | Indicates that the peer is dynamically configured. |
| sys_peer | This peer is the system peer. |
| candidate | This peer is chosen as a candidate in the combine algorithm. |
| reject | This peer is rejected by the selection algorithm. |
| falsetick | This peer is dropped as a falseticker by the selection algorithm. |
| outlier | This peer is dropped as an outlyer by the clustering algorithm. |
| stratum | The stratum number. |
| ref ID | The IPv4 address or hash of the IPv6 address of the upstream time server to which the peer is synchronized. |
| time | The last time stamp that the peer received from its master. |
| our mode | This system's mode relative to the peer (active/passive/client/server/bdcast/bdcast client). |
| peer mode | Mode of the peer relative to this system. |
| our poll intvl | This system's poll interval to this peer. |
| peer poll intvl | Poll interval of the peer to this system. |
| root delay | The delay along the path to root (the final stratum 1 time source). |
| root disp | Dispersion of the path to root. |
| reach | The peer reachability (bit string in octal). |
| delay | Round-trip delay to the peer. |
| offset | Offset of the peer clock relative to this clock. |
| dispersion | Dispersion of the peer clock. |
| precision | Precision of the peer clock. |
| version | NTP version number of the peer. |
| org time | The originate time stamp of the last packet. |
| rcv time | The receive time stamp of the last packet. |
| xmt time | The transmit time stamp of the last packet. |
| filter delay | The round-trip delay, in milliseconds, of the last 8 samples. |
| filter offset | The clock offset, in milliseconds, of the last eight samples. |
| filter error | Approximate error of the last eight samples. |

# Examples

The following is sample output from the **show ntp associations** command.

```
device# show ntp associations

address           ref               clock    st    when    poll    reach    delay    offset     disp
172.19.69.1     172.24.114.33     3         25    64         3         2.89     0.234     39377
2001:235::234
INIT 16 - 64 0 0.00 0.000 15937
* synced, # selected, + candidate, - outlayer, x falseticker, ~ configured
```

The following is sample output from the **show ntp associations detail** command.

```
device# show ntp associations detail 1o.99.40.1

10.99.40.1 configured server, candidate, stratum 3
ref ID 10.45.57.38, time d288de7d.690ca5c7 (10:33:33.1762436551 Pacific Tue Dec 06 2011)
our mode client, peer mode server, our poll intvl 10, peer poll intvl 10,
root delay 0.02618408 msec, root disp 0.10108947, reach 3, root dist 0.23610585
delay 0.92163588 msec, offset 60.77749188 msec, dispersion 70.33842156,
precision 2**-16, version 4
org time        d288defa.b260a71f (10:35:38.2992678687 Pacific Tue Dec 06 2011)
rcv time         d288defa.a2efbd41 (10:35:38.2733620545 Pacific Tue Dec 06 2011)
xmt time         d288defa.a2ae54f8 (10:35:38.2729334008 Pacific Tue Dec 06 2011)
filter delay     0.000 6.7770 6.7773 6.7711 6.7720 6.7736 6.7700 0.9921
filter offset     0.000 19.0047 19.1145 19.2245 19.3313 17.4410 15.4463 60.7777
filter disp     16000.000 16.0005 15.9975 15.9945 15.9915 15.8885 15.8855 0.0030
filter epoch     55683 55683 55685 55687 55689 55691 55693 56748
```

# show ntp status

Displays the Network Time Protocol (NTP) status.

## Syntax

**show ntp status**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface management configuration mode

NTP configuration mode

## Command Output

The **show ntp status** command displays the following information.

| Output field | Description |
|---|---|
| synchronized | Indicates that the system clock is synchronized to the NTP server or peer. |
| stratum | Indicates that the stratum number that this system is operating. Range is from 2 to 15. |
| reference clock | The IPv4 address or the first 32 bits of the MD5 hash of the IPv6 address of the peer to which the clock is synchronized. |
| precision | Precision of the clock of this system, in Hz. |
| reference time | Reference time stamp. |
| clock offset | Offset of clock (in milliseconds) to synchronized peer. |
| root delay | Total delay (in milliseconds) along the path to the root clock. |
| root dispersion | Dispersion of the root path, in milliseconds. |
| peer dispersion | Dispersion of the root path, in milliseconds. |
| system poll interval | Poll interval of the local system. |
| last clock update | Elapsed time since the router last updated its NTP information. |
| server mode | Status of the NTP server mode for this device. |
| client mode | Status of the NTP client mode for this device. |
| NTP master mode | Status of the master mode. |
| NTP master stratum | The stratum number that will be used by this device when the master is enabled and no upstream time servers are accessible. |
| panic mode | The status of the panic mode. |

# Examples

The following is sample output from the **show ntp status** command.

```
device# show ntp status

Clock is synchronized, stratum 4, reference clock is 10.20.99.174
precision is 2**-16
reference time is D281713A.80000000 (03:21:29.3653007907 GMT+00 Thu Dec 01 2011)
msec, root delay is 24.6646 msec
root dispersion is 130.3376 msec, peer dispersion is 84.3335 msec
system poll interval is 64, last clock update was 26 sec ago
NTP server mode is enabled, NTP client mode is enabled
NTP master mode is disabled, NTP master stratum is 8
NTP is not in panic mode
```

# show openflow

Displays the configured OpenFlow parameters.

## Syntax

**show openflow**

## Modes

User EXEC mode

## Command Output

The **show openflow** command displays the following information:

| Output field | Description |
|---|---|
| Administrative Status | Enable or disable status |
| Controller Type | OpenFlow 1.0 or OpenFlow 1.3 controller |
| Controller | Number of controllers |

# Examples

The following example displays the results of the **show openflow** command.

```
device#show openflow

Administrative Status:       Enabled
Controller Type:             OFV 130
Number of Controllers: 4

Controller 1:
Connection Mode:             passive, TCP
Listening Address:           0.0.0.0
Connection Port:             6633
Connection Status:           TCP_LISTENING
Role:                           Equal
Asynchronous Configuration:  Packet-in (no-match|action|invalid-ttl)
                             Port-status (add|delete|modify)
                             Flow-removed (idle-timeout|hard-timeout|delete|grp-delete)

Controller 2:
Connection Mode:             active, TCP
Controller Address:          10.25.128.243
Connection Port:             2001
Connection Status:           OPENFLOW_ESABLISHED
Role:                           Master
Asynchronous Configuration:  Packet-in (no-match|action|invalid-ttl)
                             Port-status (add|delete|modify)
                             Flow-removed (idle-timeout|hard-timeout|delete|grp-delete)
Controller 3:
Connection Mode:             active, TCP
Controller Address:          10.25.128.242
Connection Port:             6633
Connection Status:           OPENFLOW_ESABLISHED
Role:                           Slave
Asynchronous Configuration:  Port-status (add|delete|modify)

Controller 4:
Connection Mode:             active, TCP
Controller Address:          10.25.128.250
Connection Port:             2002
Connection Status:           OPENFLOW_ESABLISHED
Role:                           Slave
Asynchronous Configuration:  Port-status (add|delete|modify)

Match Capability:
Port, Destination MAC, Vlan, Vlan PCP
Openflow Enabled Ports:      1/1/1 1/1/2
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show openflow controller

Displays the controller information in a flow.

## Syntax

**show openflow controller**

## Modes

User EXEC mode

## Command Output

The **show openflow controller** command displays the following information:

| Output field | Description |
|---|---|
| Mode | Gives the active and passive connection of the controller. |
| IP address | IP address of the port |
| Port | Port number |
| Status | After the connection and OpenFlow handshake, the controller gives the role of OpenFlow channel. |
| Role | Equal, Master and Slave role for the controller. |

## Examples

The following example displays the results of the **show openflow controller** command.

```
device# show openflow controller
------------------------------------------------------------------------
Contlr Mode  TCP/SSL IP-address     Port       Status            Role
------------------------------------------------------------------------
1   (Equal)    passive  TCP    0.0.0.0          6633   TCP_LISTENING
2   (Master)   active   TCP    10.25.128.179    6633   OPENFLOW_ESABLISHED
3   (Slave)    active   TCP    10.25.128.177    6633   OPENFLOW_ESABLISHED
3   (Equal)    active   TCP    10.25.128.165    6633   OPENFLOW_ESABLISHED
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show openflow flows

Displays the flows information on the OpenFlow ports.

## Syntax

**show openflow flows**

## Modes

User EXEC mode

## Command Output

The **show openflow flows** command displays the following information:

| Output field | Description |
|---|---|
| Flow | Number of flows |
| Packet | Total Number of data packets trapped to be sent to controller |
| Byte | Total Number of data bytes trapped to be sent to controller |

# Examples

The following example displays the output for flows.

```
device# show openflow flows

Total Number of data packets sent to controller:                    0
Total Number of data bytes sent to controller  :                    0
Total Number of data packets from controller   :                    0
Total Number of data bytes from controller     :                    0

Total Number of Flows: 1
        Total Number of Port based Flows: 1
        Total Number of L2 Generic Flows: 0
        Total Number of L3 Generic Flows: 0
        Total Number of L2+L3 Generic Flows: 0
        Total Number of L23 Generic Flows: 0

Total Number of Hardware entries for flows: 1
        Total Number of Hardware entries for Port flow: 1
        Total Number of Hardware entries for Generic flow: 0

Total Number of Openflow interfaces: 6
        Total Number of L2  interfaces: 2
        Total Number of L3  interfaces: 4
        Total Number of L23 interfaces: 0

Flow ID: 2 Priority: 32768 Status: Active
        Rule:
           In Port:      e1/1/1
           Ether type:   0x800
           Destination IP:       19.0.0.19      Subnet IP:       255.255.255.255
        Instructions: Apply-Actions
                Action: FORWARD
                        Out Group:  11
        Statistics:
           Total Pkts: 0
           Total Bytes: 0
Idle and Hard timeouts:
                 Received Flow idle timeout = 0
                 Received Flow hard timeout = 0
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | This command was modified for out group and timeouts. |

# show openflow groups

For a group or a range of groups, displays the maximum number of actions in a bucket, the maximum number of buckets in a group, and the maximum number of groups.

## Syntax

**show openflow groups** [ *group-id* ]

**show openflow groups** *group-id* **to** *group-id*

## Parameters

**groups** *group-id*

Displays details of an OpenFlow group or range of groups.

**to**

Indicates a range of groups.

## Modes

User EXEC mode

## Command Output

The **show openflow groups** command displays the following information:

| Output field | Description |
|---|---|
| Group | Maximum number of groups in a flow |
| Bucket | Number of buckets per group |
| Action | Number of actions per bucket |

# Examples

The following example displays the output from the **show openflow groups** command.

```
device#show openflow groups

Max number of groups             : 512
Max number of buckets per group  :  64
Max number of actions per bucket :   6

Max number of SELECT groups          : 512
Max number of buckets in SELECT group: 32
Starting Trunk ID for SELECT groups  : 49664
Group id 11

  Transaction id        410
  Type                  SELECT
  Packet Count          0
  Byte   Count          0
  Flow   Count          1
  Number of buckets     2
bucket #1
    Weight             1
    Number of actions   5
      action 1: out port: 1/1/2
      action 2: Dec IP TTL
      action 3: VLAN: 1111
      action 4: Source MAC: 0011.1111.1111
      action 5: Destination MAC: 0022.2222.2222


  bucket #2
    Weight             1
    Number of actions   5
      action 1: out port: 1/1/17
      action 2: Dec IP TTL
      action 3: VLAN: 1122
      action 4: Source MAC: 0033.3333.3333
      action 5: Destination MAC: 0044.4444.4444

  Forwarding information:
    Select Index: 49664


----

Total no. of entries printed: 1
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |
| 08.0.50 | This command was modified to show action list. |

# show overlay-gateway

Displays information for all overlay-gateways, a specific gateway, a VLAN, VNI, or remote site.

## Syntax

**show overlay-gateway** [ *gateway-name* ] [ **detail** ]

**show overlay-gateway** [ *gateway-name* ] [ **vlan** *vlan-id* ]

**show overlay-gateway** [ *gateway-name* ] [ **vni** *vni-id* ]

## Parameters

*gateway-name*
Specifies the VXLAN gateway name for which information is displayed.

**detail**
Displays more extensive information for the overlay-gateway or gateways.

**vlan** *vlan-id*
Displays information for the designated extended VLAN.

**vni** *vni-id*
Displays information for the designated VXLAN Network Identifier

**site** *site-name*
Displays information for the remote site specified.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

## Command Output

The **show overlay-gateway** command set displays the following information:

| Output field | Description |
|---|---|
| Overlay Gateway Name | Name configured for the overlay-gateway |
| Type | Type of gateway (layer2-extension) |
| Source IP interface | Loopback interface number, VRF used, and local IPv4 address of the source VEP |
| Total Mapped Vlans | Number of VLANS mapped to the VNI associated with the overlay-gateway name |
| Total Sites | Number of tunnels associated with the overlay-gateway |
| Total Overlay Gateways | Number of overlay-gateways configured in the network |
| VLAN-ID | VLAN number for a VLAN extended over the overlay-gateway |
| VN-ID | VXLAN Network Identifier associated with the VLAN |

| Output field | Description |
|---|---|
| VFI | Layer 2 VPN identifier in the hardware (should match the VLAN ID) |
| Access-Port | Total number of local ports belonging to the mapped VLAN |
| Extended-Site | Number of remote sites to which the mapped VLAN is extended |
| SiteName | Name of remote site |
| IP-Address | IP address of the remote site |
| Status | Status of the remote site (Up, Down) |
| Ext-Vlans | Mapped VLANS that are extended to the remote site |

# Examples

The following example shows sample output for the **show overlay-gateway** command. Only one overlay-gateway, sanjose, is configured.

```
ICX7750-48F Router# show overlay-gateway

Overlay Gateway Name    : sanjose
Type                    : layer2-extension
Source IP Interface     : loopback 1 (vrf: default-vrf, IP address: 7.7.7.7)
Total Mapped Vlans      : 2
Total Sites             : 1

Total 1 Overlay Gateways
```

The following example provides detail on the overlay-gateway "sanjose." The output includes a list of VLANS, their associated VNI, VFI, access port, and VXLAN tunnel. The name of the remote site ("denver") is given, along with the remote IP address, its status, and the connected extended VLANs at the remote end.

```
ICX7750-48F Router# show overlay-gateway sanjose detail
Overlay Gateway Name    : sanjose
Type                    : layer2-extension
Source IP Interface     : loopback 1 (vrf: default-vrf, IP address: 7.7.7.7)
Total Mapped Vlans      : 2
Total Sites             : 1
#    VLAN-ID    VN-ID    VFI     Access-Port Extended-Site
-    -------    -----    ---     ----------- -------------
1    101        25838    101     5           1
2    102        67924    102     3           1
#    SiteName                                        IP-Address       Status Ext-Vlans
-    --------                                        ----------       ------ ---------
1    denver                                          2.2.2.2          Up     (101,102)
```

The following example displays information for overlay-gateway sanjose VLAN 101 and VLAN 102.

```
ICX7750-48F Router# show overlay-gateway sanjose vlan 101
Overlay Gateway Name    : sanjose
#    VLAN-ID    VN-ID    VFI     Access-Port Extended-Site
-    -------    -----    ---     ----------- -------------
1    101        25838    101     5           1

ICX7750-48F Router# show overlay-gateway sanjose vlan 102
Overlay Gateway Name    : sanjose
#    VLAN-ID    VN-ID    VFI     Access-Port Extended-Site
-    -------    -----    ---     ----------- -------------
1    102        67924    102     3           1
```

The following example displays information for two VNIs on the same overlay-gateway.

```
ICX7750-48F Router# show overlay-gateway sanjose vni 25838
Overlay Gateway Name    : sanjose
#    VN-ID      VLAN-ID  VFI     Access-Port Extended-Site
-    -----      -------  ---     ----------- -------------
1    25838      101      101     5           1

ICX7750-48F Router# show overlay-gateway sanjose vni 67924
Overlay Gateway Name    : sanjose
#    VN-ID      VLAN-ID  VFI     Access-Port Extended-Site
-    -----      -------  ---     ----------- -------------
1    67924      102      102     3           1
```

The following example displays information for remote site denver. Its status is up.

```
ICX7750-48F Router# show overlay-gateway sanjose site denver
Overlay Gateway Site Name     : denver
IP address                    : 2.2.2.2
Status                        : Up
Extended Vlans                :
   101, 102
Total 2 Extended Vlan
```

The following example displays information for the same remote site, which is down because no source interface is configured.

```
ICX7750-48F Router# show overlay-gateway sanjose site denver
Overlay Gateway Site Name     : denver
IP address                    : 2.2.2.2
Status                        : Down(No Source Interface)
Extended Vlans                :
   101, 102
Total 2 Extended Vlan
```

The following example shows information for the same remote site, which is down because there is no route to the destination.

```
ICX7750-48F Router# show overlay-gateway sanjose site denver
Overlay Gateway Site Name     : denver
IP address                    : 2.2.2.2
Status                        : Down(No Route to Destination)
Extended Vlans                :
   101, 102
Total 2 Extended Vlan
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | The command was introduced for ICX 7750 devices. |

# show openflow interface

Displays the information about the interfaces in a OpenFlow flow.

## Syntax

**show openflow interface**

## Modes

User configuration mode

## Usage Guidelines

The **show openflow interface** command displays the physical port, up and down links, tag status, MAC addresses, and the modes.

## Command Output

The **show openflow interface** command displays the following information:

| Output field | Description |
|---|---|
| Port | Port Number |
| Link | Link status |
| Speed | Configured speed |
| Tag | Tag status |
| Mac Address | MAC address of the port |
| Mode | Gives the information about the layers |

## Examples

The following example displays information for all OpenFlow interfaces.

```
device# openflow enable layer3 hybrid
device# show openflow interface

Total number of Openflow interfaces: 5

Port  Link     Speed Tag MAC            OF-portid Name    Mode
1/1/1   Up       1G    Yes 000c.dbf5.bd00 1                Layer2
1/1/2   Up       1G    Yes 000c.dbf5.bd01 2                Layer2
1/1/3   Up       1G    Yes 000c.dbf5.bd01 3                Hybrid-Layer3
1/1/4   Up       1G    Yes 000c.dbf5.bd01 4                Hybrid-Layer3
1/1/5   Up       1G    Yes 000c.dbf5.bd01 5                Hybrid-Layer3
```

The following command displays information for a particular interface on a specific slot and port.

```
device# show interface ethernet 1/1/6

GigabitEthernet1/1/6 is up, line protocol is up
  Port up for 51 minutes 53 seconds
  Hardware is GigabitEthernet, address is 748e.f8e7.d901 (bia 748e.f8e7.d901)
  Configured speed auto, actual 1Gbit, configured duplex fdx, actual fdx
  Configured mdi mode AUTO, actual MDI
  Member of L2 VLAN ID 100, port is untagged, port state is FORWARDING
  BPDU guard is Disabled, ROOT protect is Disabled, Designated protect is Disabled
  Link Error Dampening is Disabled
  STP configured to ON, priority is level0, mac-learning is enabled
  OpenFlow enabled, Openflow Index 1, Flow Type Layer2
  Flow Control is config enabled, oper enabled, negotiation disabled
  Mirror disabled, Monitor disabled
  Not member of any active trunks
  Not member of any configured trunks
  No port name
  Inter-Packet Gap (IPG) is 96 bit times
  MTU 1500 bytes, encapsulation ethernet
  300 second input rate: 3904 bits/sec, 7 packets/sec, 0.00% utilization
  300 second output rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  23153 packets input, 1530094 bytes, 0 no buffer
  Received 1721 broadcasts, 21432 multicasts, 0 unicasts
  0 input errors, 0 CRC, 0 frame, 0 ignored
  0 runts, 0 giants
  0 packets output, 0 bytes, 0 underruns
  Transmitted 0 broadcasts, 0 multicasts, 0 unicasts
  0 output errors, 0 collisions
  Relay Agent Information option: Disabled

Egress queues:
Queue counters    Queued packets    Dropped Packets
    0                   0                 0
    1                   0                 0
    2                   0                 0
    3                   0                 0
    4                   0                 0
    5                   0                 0
    6                   0                 0
    7                   0                 0
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show openflow meters

Displays all the meters in a OpenFlow flow.

## Syntax

**show openflow meters** [ *meter-id* ]

## Parameters

**meters** *meter-id*

Shows details of a specific OpenFlow meter.

## Modes

User EXEC mode

## Command Output

The **show openflow meters** command displays the following information:

| Output field | Description |
|---|---|
| Meter-id | Meter number |
| Band | Number of bands in a meter |
| Band type | Band type ( supported type: Drop, DSCP_REMARK) |
| Rate | Rate of the band |
| Counter | Band specific counter |

## Examples

The following example displays output with single meter band.

```
device(config)# show openflow meters 1
Meter id: 1

  Transaction id:        1437
  Meter Flags:            KBPS BURST STATS
  Flow Count:            0
  Number of  bands:      1
  In packet count:       -NA-
  In byte count:         0

  Band Type:    DROP

    Rate:                        750000
    Burst size:                  1500          kb
    In packet band count:        -NA-
    In byte band count:          0
```

The following example displays output with two meter bands.

```
device(config)# show openflow meters 2
Meter id: 2

  Transaction id:        1438
  Meter Flags:            KBPS BURST STATS
  Flow Count:            0
  Number of  bands:      2
  In packet count:       -NA-
  In byte count:         0

  Band Type:    DSCP-REMARK

    Rate:                          750000
    Burst size:                    1500          kb
    Prec level:                    1
    In packet band count:          -NA-
    In byte band count:            0


  Band Type:    DROP

    Rate:                          1000000
    Burst size:                    2000          kb
    In packet band count:          -NA-
    In byte band count:            0
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show optic

Displays the optic temperature and power information for qualified QSFP+, SFP, or SFP+ transceivers installed in a device.

## Syntax

**show optic** [ **threshold** ] *unit/slot/port*

## Parameters

**threshold**

Displays the thresholds for a qualified optical transceiver for the specified port.

*unit/slot/port*

Displays optics information for the QSFP+, SFP, or SFP+ transceiver in the specified interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

NTP configuration mode

## Usage Guidelines

The **show optic** command takes advantage of information stored and supplied by the manufacturer of the QSFP+, SFP, or SFP+ transceiver. This information is an optional feature of the Multi-Source Agreement standard defining the optical interface. Not all component suppliers have implemented this feature set. When the QSFP+, SFP, or SFP+ transceiver does not supply the information, a "Not Available" message is displayed for the specific port on which the module is installed.

## Command Output

The **show optic** command displays the following information.

| Output field | Description |
|---|---|
| Temperature | The operating temperature, in degrees Celsius, of the optical transceiver, followed by the alarm status. |
| Tx Power | The transmit power signal, in decibels (dB), of the measured power referenced to one milliwatt (mW), followed by the alarm status. |
| Rx Power | The receive power signal, in decibels (dB), of the measured power referenced to one milliwatt(mW), followed by the alarm status. |
| Tx Bias Current | The transmit bias power signal, in milliamperes (mA), followed by the alarm status. |

For Temperature, Tx Power, Rx Power, and Tx Bias Current in the **show optic** command output, values are displayed along with one of the following alarm status values: Low-Alarm, Low-Warn, Normal, High-Warn, or High-Alarm. The thresholds that determine these status values are set by the manufacturer of the optical transceivers. The following table describes each of these alarm status values.

**TABLE 12** Alarm status values

| Status value | Description |
|---|---|
| Low-Alarm | The monitored level has dropped below the "low-alarm" threshold set by the manufacturer of the optical transceiver. |
| Low-Warn | The monitored level has dropped below the "low-warn" threshold set by the manufacturer of the optical transceiver. |
| Normal | The monitored level is within the "normal" range set by the manufacturer of the optical transceiver. |
| High-Warn | The monitored level has climbed above the "high-warn" threshold set by the manufacturer of the optical transceiver. |
| High-Alarm | The monitored level has climbed above the "high-alarm" threshold set by the manufacturer of the optical transceiver. |

# Examples

The following sample output displays optics information for the specified interface if you are displaying for a port equipped with SFP or SFP+ transceivers.

```
device(config)# show optic 2/1/1
 Port    Temperature     Tx Power      Rx Power       Tx Bias Current
+-----+-------------+-------------+--------------+---------------------+
2/1/1    32.2578 C    -002.5157 dBm  -002.8091 dBm    5.966 mA
         Normal       Normal         Normal           Normal
```

This example displays optics information for a specified interface if you are displaying for a port equipped with QSFP+ transceivers, where there are 4 TX bias and 4 RX channels or lanes.

```
device(config)# show optic 1/2/6

                        40GBASE_SR4
                       ===============
 Port  Temperature    Tx Power       Rx Power        Tx Bias Current
1/2/6  37.4531 C     005.1838 dBm  -002.1752 dBm    7.154 mA
       Normal        Normal         Normal          Normal

 Chan  Rx Power #1    Rx Power #2     Rx Power #3     Rx Power #4
+-----+--------------+---------------+---------------+---------------+
       -002.1752 dBm   -003.1704 dBm   -001.4466 dBm   -001.6241 dBm
        Normal          Normal          Normal          Normal

 Chan  Tx Bias #1   Tx Bias #2     Tx Bias #3     Tx Bias #4
+----+-----------+-------------+-------------+---------------+
       7.154 mA      6.962 mA      6.972 mA      7.014 mA
       Normal        Normal        Normal        Normal
```

The following sample output displays the thresholds for the qualified optical transceiver for the specified port

```
device(config)# show optic threshold 2/1/1
show optic thresholds 2/1/1
Port 2/1/1 sfp monitor thresholds:
Temperature High alarm             5a00     90.0000 C
Temperature Low alarm              fb00     -5.0000 C
Temperature High warning           5500     85.0000 C
Temperature Low warning            0000      0.0000 C
TX Bias High alarm                 1482     10.500  mA
TX Bias Low alarm                  04e2      2.500  mA
TX Bias High warning               1482     10.500  mA
TX Bias Low warning                04e2      2.500  mA
TX Power High alarm                4e20    003.0102 dBm
TX Power Low alarm                 04ec   -008.9962 dBm
TX Power High warning              1edc   -001.0237 dBm
TX Power Low warning               0c62   -004.9894 dBm
RX Power High alarm                4e20    003.0102 dBm
RX Power Low alarm                 013b   -015.0168 dBm
RX Power High warning              1edc   -001.0237 dBm
RX Power Low warning               013b   -015.0168 dBm
```

# History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was introduced. |

# show optic-timer

Displays the digital optical monitoring (DOM) time interval setting.

## Syntax

**show optic-timer** [ *unit/slot/port* ]

## Parameters

*unit/slot/port*

Specifies a particular Ethernet port.

## Modes

Global configuration mode

## Usage Guidelines

## Examples

The following example displays the DOM time interval setting.

```
device(config)# show optic-timer
Optical monitoring timer interval is 8 mins
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show packet-inerror-detect

Displays details related to the monitoring for inError packets for configured ports.

## Syntax

**show packet-inerror-detect**

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

Use this show command to view details related to the monitoring of inError packets for configured ports.

## Command Output

The **show packet-inerror-detect** command displays the following information:

| Output field | Description |
|---|---|
| Sampling interval | Displays the configured sampling interval. |
| Port | Identifies a port. |
| Packet inError count | The number of inError packets received in the sampling interval for the specific port. |
| State | Displays the status for the specific port. |

## Examples

The following example displays details related to the monitoring for inError packets for configured ports.

```
device# show packet-inerror-detect

Sampling interval 5 secs

Port    Packet inError count State
1/1/1   30                   Operational
1/1/37  10                   ERR-DISABLED
2/1/1   100                  Operational
```

## History

| Release version | Command history |
|---|---|
| 07.3.00g | This command was introduced. |

# show pki

Displays information on PKI, including information on certificates and other options.

## Syntax

**show pki** { **certificates** { **local** | **trustpoint** } | **counters** | **crls** | **enrollment-profile** | **entity** | **key** | **logging-statistics** | **trustpoint** }

## Parameters

**certificates**
Displays PKI certificates.

**counters**
Displays PKI counters.

**crls**
Displays the PKI certification revocation list if there is one.

**enrollment-profile**
Displays PKI enrollment profile.

**entity**
Displays PKI entity.

**key**
Displays router public keys.

**logging-statistics**
Displays PKI logging statistics.

**trustpoint**
Displays PKI trustpoint information.

## Modes

User EXEC mode

## Examples

The following example shows output for the **show pki certificates local** command.

```
device# show pki certificates local
----------------PKI LOCAL CERTIFICATE ENTRY-----------------
CA: trustRSA
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 4100 (0x1004)
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=IN, ST=KA, L=Bangalore, O=Ruckus Arris, OU=NEBU, CN=ROOT RSA
        Validity
            Not Before: Feb 23 16:19:43 2018 GMT
            Not After : Feb 21 16:19:43 2028 GMT
        Subject: CN=ICX RSA, ST=KA, C=IN, O=NEBU, OU=Ruckus Arris
```

The following example shows output for the **show pki certificates trustpoint** command.

```
device# show pki certificates trustpoint
----------------PKI TRUSTPOINT CERTIFICATE ENTRY-----------------
CA: trustRSA
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            e2:11:82:3f:37:c2:6f:c0
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=IN, ST=KA, L=Bangalore, O=Ruckus Arris, OU=NEBU, CN=ROOT RSA
        Validity
            Not Before: Feb 23 05:38:11 2018 GMT
            Not After : Feb 23 05:38:11 2023 GMT
        Subject: C=IN, ST=KA, L=Bangalore, O=Ruckus Arris, OU=NEBU, CN=ROOT RSA
```

The following example shows output for the **show pki counters** command.

```
device# show pki counters
------------PKI-COUNTERS----------
PKI Sessions Started: 3701
PKI Sessions Ended: 3701
PKI Sessions Active: 0
Successful Validations: 35
Failed Validations: 3855
Bypassed Validations: 0
Pending Validations: 5
CRLs checked: 0
CRL - fetch attempts: 0
CRL - failed attempts: 0
```

The following example shows output for the **show pki enrollment-profile** command.

```
device# show pki enrollment-profile
----------------PKI ENROLLMENT PROFILE ENTRY-----------------
  Enrollment Profile: profile1
  Authentication Command: WINN6C3R0LUDAJ.
  Authentication URL: http://WINN6C3R0LUDAJ.
  Enrollment URL: http://ipfvt-mylab.englab.brocade.com/CertSrv/mscep/mscep.dll
  SCEP password: helloutthere
```

The following example shows output for the **show pki entity** command.

```
device# show pki entity
----------------PKI ENTITY ENTRY-----------------
  Entity Name: spatha27
    Common Name: Spatha
    Organization Name: SQA
    Organization Unit Name: ICX
    State Name: KA
    Country Name: IN

----------------PKI ENTITY ENTRY-----------------
  Entity Name: ent1
    Common Name: en1
    State Name: KA
    Country Name: IN
    Location: BLR

----------------PKI ENTITY ENTRY-----------------
  Entity Name: entity1
    Common Name: tester1
    Organization Name: BRCD
    Organization Unit Name: FI
    State Name: BC
    Country Name: CA
    Email: user@brocade.com
    Location: BG
```

The following example shows output for the **show pki key** command.

```
device#  show pki key mypubkey  all
----------------PKI PUBLIC KEY ENTRY-----------------
 Public key of generated EC key pair:
  The key label is marcia_ec
  Public-Key: (384 bit)
  pub:
      04:61:f6:d4:bf:e0:85:8f:2f:70:e3:79:36:d9:22:
      98:ca:3e:6e:10:a3:cd:b9:0a:e9:2d:26:ce:a3:fc:
      96:c5:04:f7:28:6b:fa:fb:e1:36:51:4b:05:05:95:
      da:e7:14:5f:59:68:16:2b:fc:c7:a0:d6:a0:72:85:
      28:dd:54:10:1e:42:51:0d:8e:d7:6b:2f:92:cc:e2:
      ac:f6:f5:89:64:da:54:af:b5:26:e1:f6:a5:25:f2:
      a9:93:3c:9a:b8:93:5b
  ASN1 OID: secp384r1
```

The following example shows output for the **show pki logging-statistics** command.

```
device# show pki logging-statistics
------------------------PKI logging statistics--------------------------------
Type of packet:                      |    TX_PACKETS      |     RX_PACKETS
-------------------------------------|--------------------|-----------------
enrollment packets:                  |       0            |          0
authentication packets:              |       1            |          1
revocation check packets:            |       116          |          0
peer certificate download packets:   |       0            |          0
certificate imports through http:    |       0            |          0
Note:enrollment packets can be 2x of actual enrollments depends on server settings
```

The following example shows output for the **show pki trustpoint** command.

```
device# show pki trustpoint
----------------PKI TRUSTPOINT ENTRY-----------------
  CA: trustRSA
  Key Information:
    The key label is icx_rsa_key
    Public-Key: (2048 bit)
    Modulus:
        00:c5:81:6f:98:aa:f8:e4:a8:2d:d9:f3:d7:d0:e7:
        5e:be:59:4b:4c:d0:c9:aa:a8:53:82:dd:2f:df:09:
        c1:78:c5:38:63:c3:d7:73:47:ed:43:6c:d6:d1:ed:
        99:82:e7:51:c6:03:bc:8e:8f:97:e5:1b:b5:71:a1:
        46:f4:a8:b2:bb:6e:61:54:e2:42:1e:63:f8:79:78:
        6b:bd:d8:63:67:c1:b7:6f:78:cc:9d:16:42:df:81:
        d2:98:24:2b:70:60:10:ec:0e:5c:d9:be:7e:e1:a0:
        27:b8:e0:65:73:99:de:18:59:05:e7:7e:df:f1:1e:
        ac:ab:33:7a:7e:6e:d5:99:85:95:fc:c8:a7:1f:c3:
        d2:43:74:2e:c6:15:80:b6:fc:73:4c:23:30:2a:c1:
        26:d0:84:4c:58:96:0b:4c:1c:f0:87:cf:d3:28:68:
        0a:65:f7:fd:33:cb:92:c7:d5:8d:df:7b:9b:03:92:
        d8:75:03:1c:f6:1b:09:b3:6d:3c:2a:7e:6a:02:10:
        21:5c:46:87:46:73:57:7c:66:8f:a4:bb:a4:6b:ae:
        30:d2:63:a0:44:44:6b:48:e2:ab:8e:fa:d4:d7:f7:
        30:87:c1:11:ac:22:9f:e9:10:52:ee:22:70:c6:f7:
        6b:5b:eb:7f:f3:b3:01:a9:d6:25:10:97:1b:9d:7e:
        50:51
    Exponent: 65537 (0x10001)
  Configured Fingerprint for authentication:
    D8:BC:F5:94:BA:72:9D:F3:34:77:FD:AA:5B:A2:FD:B6:59:A3:00:27
  Enrollment Protocol:SCEP

----------------PKI TRUSTPOINT ENTRY-----------------
  CA: trust1
  Entity Name: entity1
    Common Name: tester1
    Organization Name: BRCD
    Organization Unit Name: FI
    State Name: BC
    Country Name: CA
    Email: user@brocade.com
    Location: BG
  Configured Fingerprint for authentication:
    d2:52:b6:5a:1d:a2:95:3b:f4:e6:05:33:84:05:97:16:75:15:bf:04
  Enrollment Protocol:SCEP
  Enrollment Profile: profile1
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show pod

Displays Ports on Demand (PoD) licensing information.

## Syntax

**show pod** [ **unit** *unit_id*]

## Parameters

**unit** *unit_id*

Indicates the PoD unit ID number. The *unit_id* can be from 1 through 12 on the Ruckus ICX 7250 devices.

## Modes

Privileged EXEC level.

## Usage Guidelines

The command displays PoD license configuration for all ports in a stack unit. The command is supported on Ruckus ICX 7250 and Ruckus ICX 7150devices.

On the 24-port and 48-port models of the Ruckus ICX 7150, the PoD ports are 1/3/1 to 1/3/4. On the Ruckus ICX 7150-C12 model, the PoD ports are 1/3/1 and 1/3/2.

## Command Output

The **show pod** command displays the following information:

| Output field | Description |
|---|---|
| Unit-Id | The unit ID number of the PoD port. |
| PoD license capacity | The port capacity of the PoD license that is purchased. |
| PoD license capacity used | The number of PoD ports that are upgraded to 10 Gbps port speed. |
| PoD-ports | The list of PoD ports in the PoD unit. |
| Lic-Available | Displays whether the license is available for the port. |
| Lic-Used | Displays whether the license is used by the port. |

# Examples

The following **show pod** command example output displays PoD licensing information

```
device#show pod
Unit-Id: 1
PoD license capacity:  8
PoD license capacity used:   8

PoD-ports    Lic-Available Lic-Used
1/2/1      Yes             Yes
1/2/2      Yes             Yes
1/2/3      Yes             Yes
1/2/4      Yes             Yes
1/2/5      Yes             Yes
1/2/6      Yes             Yes
1/2/7      Yes             Yes
1/2/8      Yes             Yes

Unit-Id: 11
PoD license capacity:  8
PoD license capacity used:   8

PoD-ports    Lic-Available Lic-Used
11/2/1     Yes             Yes
11/2/2     Yes             Yes
11/2/3     Yes             Yes
11/2/4     Yes             Yes
11/2/5     Yes             Yes
11/2/6     Yes             Yes
11/2/7     Yes             Yes
11/2/8     Yes             Yes
```

# History

| Release version | Command history |
|---|---|
| 07.3.00 | This command was introduced. |

# show port security

Displays the port security information.

## Syntax

**show port security** [ **ethernet** *stack/slot/port* [ **restricted-macs** ] ]

**show port security mac** [ **ethernet** *stack/slot/port* | **unit** *stack-unit-num* ]

**show port security statistics** [ **ethernet** *stack/slot/port* | **unit** *stack-unit-num* [ **brief** ] ]

## Parameters

**ethernet** *stack/slot/port*
Specified Ethernet interface.

**restricted-macs**
Displays information about restricted MAC addresses on the specified port.

**mac**
Displays secure MAC addresses configured on a device.

**unit** *stack-unit-num*
Specifies the stack unit number.

**statistics**
Displays port security statistics.

**brief**
Displays brief information.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Port security configuration mode

Port security interface configuration mode

## Usage Guidelines

The **show port security** command without any options displays the port security settings for all the ports.

## Command Output

The **show port security ethernet** command displays the following information:

| Output field | Description |
|---|---|
| Port | The slot and port number of the interface. |
| Security | Whether port security has been enabled on the interface. |
| Violation | The action to be undertaken when a security violation occurs, either "shutdown" or "restrict". |
| Shutdown-Time | The number of seconds a port is shut down following a security violation, if the port is set to "shutdown" when a violation occurs. |
| Age-Time | The amount of time, in minutes, MAC addresses learned on the port will remain secure. |
| Max-MAC | The maximum number of secure MAC addresses that can be learned on the interface. |

The **show port security mac** command displays the following information:

| Output field | Description |
|---|---|
| Port | The slot and port number of the interface. |
| Num-Addr | The number of MAC addresses secured on this interface. |
| Secure-Src-Addr | The secure MAC address. |
| Resource | Whether the address was secured using a local or global resource. |
| Age-Left | The number of minutes the MAC address will remain secure. |
| Shutdown/Time-Left | Whether the interface has been shut down due to a security violation and the number of seconds before it is enabled again. |

**NOTE**
After every switchover or failover, the MAC "Age-Left" timer is reset to start because it is not synchronized between the master and the standby stack unit.

The **show port security statistics** command displays the following information:

| Output field | Description |
|---|---|
| Port | The slot and port number of the interface. |
| Total-Addrs | The total number of secure MAC addresses on the interface. |
| Maximum-Addrs | The maximum number of secure MAC addresses on the interface. |
| Violation | The number of security violations on the port. |
| Shutdown/Time-Left | Whether the port has been shut down due to a security violation and the number of seconds before it is enabled again. |

# Examples

The following example displays the port security settings for port 1/1/1.

```
device# show port security ethernet 1/1/1
Port  Security Violation Shutdown-Time Age-Time  Max-MAC
----- -------- --------- ------------- --------- -------
1/1/1 disabled shutdown   10             10        1
```

The following example shows the list of secure MAC addresses configured on the device.

```
device# show port security mac
Port   Num-Addr  Secure-Src-Addr  Resource  Age-Left  Shutdown/Time-Left
-----  --------  ---------------  --------  --------- ------------------
1/1/1   1        0000.018.747c     Local     10          no
```

The following example displays port security statistics for interface 1/1/1.

```
device# show port security statistics ethernet 1/1/1
Port   Total-Addrs Maximum-Addrs Violation Shutdown/Time-Left
-----  ----------- ------------- --------- ------------------
1/1/1   1          1               0          no
```

# show power-savings-statistics

Displays the power savings statistics for the device.

## Syntax

**show power-savings-statistics**

## Modes

Global configuration mode

## Usage Guidelines

## Examples

The following example displays the power savings statistics for the device.

```
device(config)# show power-savings-statistics

Warning - The below is a theoretical calibrated estimation, there may be +- 5% deviation  on the data.


The Power statistics of the switch for the last 5 minutes is


The total power consumption of the switch for the  past 5 minutes  is ------------------------> 76064
milli Watts

The total power savings after enabling EEE for the past 5 minutes  is ------------------------> 3598
milli Watts

The power efficiency of the Switch after Enabling EEE for the past 5 min is ------------------> 4%

                        The Port specific statistics for the past 5 minutes is


Port   EEE-State      Traffic           Power_Rating    Power_Consumed  Power_Conserved
Power_Efficiency
                  Port Utilization%        in mW             in mW          in mW               in%

1/1/1   Enable           0                333               7              257                  77
1/1/2   Enable           0                 33               76             257                  77
1/1/3   Enable           0                333               76             257                  77
1/1/4   Enable           0                333               76             257                  77
1/1/5   Enable           0                333               76             257                  77
1/1/6   Enable           0                333               76             257                  77
1/1/13  Enable           0                333               76             257                  77
1/1/14  Enable           0                333               76             257                  77
1/1/15  Enable           0                333               76             257                  77
1/1/16  Enable           0                333               76             257                  77
1/1/21  Enable           0                333               76             257                  77
1/1/22  Enable           0                333               76             257                  77
1/1/23  Enable           0                333               76             257                  77
1/1/24  Enable           0                333               76             257                  77
1/2/1   Enable           0                  0               0                0                   0
1/2/2   Enable           0                  0               0                0                   0
1/2/3   Enable           0                  0               0                0                   0
1/2/4   Enable           0                  0               0                0                   0
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# show priority-flow-control

Displays the priority flow control (PFC) on the system.

## Syntax

**show priority-flow-control**

## Modes

Privileged EXEC mode

## Examples

The following example shows the PFC status of all priority groups.

```
Device# show priority-flow-control

Global PFC Status: Enabled
PFC Enabled on PG0
PFC Disabled on PG1
PFC Disabled on PG2
PFC Disabled on PG3
```

The following example shows the PFC status disabled.

```
Device# show priority-flow-control

Global PFC Status: Disabled
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# show protected-port

Displays the system-wide configuration status of protected ports.

## Syntax

**show protected-port**

## Modes

Privileged EXEC mode

## Examples

The following example displays the system-wide status of protected ports.

```
device# show protected-port
  System-Wide Protected Ports: ethe 1/1/1 ethe 2/1/1 ethe 3/1/1
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show pvlan

Displays the PVLAN information.

## Syntax

**show pvlan** [ *vlan-id* ]

## Parameters

*vlan-id*

Displays the information for the PVLAN with the specified VLAN ID.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

## Usage Guidelines

If the VLAN ID is not specified, the command displays the default VLAN ID information. The **show pvlan** command is not supported on software-forwarding platforms.

This command displays the PVLAN configuration with respect to the primary VLAN and its associated secondary VLANs and to display the member ports, promiscous ports, and inter-switch link ports of a PVLAN.

## Examples

The following example displays sample output of the **show pvlan** command.

```
device# show pvlan
PVLAN: primary VLAN 100
 Port 1/1/4 1/1/10 1/1/11
Community VLAN 102
 Port 1/1/1 1/1/2 1/1/10 1/1/11
 Promiscuous Port: 1/1/4
 Inter switch link Port: 1/1/10 1/1/11
 BpduGuard enabled Port: 1/1/1 1/1/2
Isolate VLAN 101
 Port 1/1/3 1/1/10 1/1/11
 Promiscuous Port: 1/1/4
 Inter switch link Port: 1/1/10 1/1/11
 BpduGuard enabled Port: 1/1/1 1/1/2
```

# show pvstplus-protect-ports

Displays the status of the PVST+ Protect feature, configured by means of the **pvstplus-protect** command.

## Syntax

**show pvstplus-protect-ports** [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ]

## Parameters

**ethernet**

Specifies an Ethernet port.

*unit/slot/port*

Number of an Ethernet port. Ranging is allowed by means of the **to** keyword.

to

Enables optional ranging.

## Modes

Privileged EXEC mode

## Examples

The following example displays the status of PVST+ Protect on all Ethernet interfaces, including the number of dropped PVST+ BPDUs.

```
device# show pvstplus-protect-ports
        Port    PVST Drop Count
        1/1/1   11
        1/1/2   0
        1/1/3   0
        1/1/4   0
```

The following example displays the status of PVST+ Protect on a single Ethernet interface.

```
device# show pvstplus-protect-ports ethernet 1/1/1
PVST-protect is enabled on port 1/1/1.  PVST drop count is 11
```

The following example displays the status of PVST+ Protect on a range of Ethernet interfaces.

```
device# show pvstplus-protect-ports ethernet 1/1/1 to 1/1/4
```

## History

| Release version | Command history |
|---|---|
| 08.0.30mb | This command was introduced. |

# show qd-buffer-profile

Displays the user-configurable buffer profile configuration on the device.

## Syntax

**show qd-buffer-profile** { *profile-name* | **all** }

## Parameters

*profile-name*

Displays the user-configurable buffer profile for a specific buffer profile.

**all**

Displays all the user-configurable buffer profiles on the device.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show qd-buffer-profile** command displays the following information:

| Output field | Description |
|---|---|
| User Buffer Profile | The name of the user-configurable buffer profile. |
| Port-type | The type of the port: 1 Gbps or 10 Gbps or All. |
| Total Buffers | The total number of buffers allocated to the port. |
| Total Descriptors | The total number of descriptors allocated to the port. |
| Per Queue details | The names of the queues. |
| Buffers | The total number of buffers allocated to the queue. |
| Descriptors | The total number of descriptors allocated to the queue. |

# Examples

The following example displays sample output of the **show qd-buffer-profile** command.

```
device(config)# show qd-buffer-profile OneGigProfile
User Buffer Profile: OneGigProfile Port-type: 1Gig
Total Buffers = 8096 Total Descriptors = 8096
Per Queue details:  Buffers   Descriptors
Traffic Class 0     50        38
Traffic Class 1     50        38
Traffic Class 2     50        38
Traffic Class 3     50        38
Traffic Class 4     50        38
Traffic Class 5     50        38
Traffic Class 6     132       132
Traffic Class 7     20        20
```

# show qos egress-buffer-profile

Displays information about egress buffer profiles.

## Syntax

**show qos egress-buffer-profile** [ *user-profile-name* | **all** ]

## Parameters

*user-profile-name*

Displays information for the specified egress buffer profile.

**all**

Displays information for all egress buffer profiles configured in the system and a list of all ports attached to any egress buffer profile.

## Modes

Global configuration mode

## Usage Guidelines

For the Ruckus ICX 7150 device, this command displays the share port level of the egress buffer profile.

For the Ruckus ICX 7250, ICX 7450, and ICX 7750 devices, this command displays the share queue level of the egress buffer profile.

## Examples

On a Ruckus ICX 7250, ICX 7450, or ICX 7750 device, the following example displays information for an egress buffer profile named egress1.

```
Device(config)# show qos egress-buffer-profile egress1

Egress Buffer Profile: egress1
Ports attached:  1/1/2
Per Queue Details:       Share Level:
Queue 0                  level4-1/9
Queue 1                  level3-1/16
Queue 2                  level3-1/16
Queue 3                  level3-1/16
Queue 4                  level3-1/16
Queue 5                  level3-1/16
Queue 6                  level3-1/16
Queue 7                  level2-1/32
```

On a Ruckus ICX 7150, the following example displays information for an egress buffer profile named egress2.

```
device# show qos egress-buffer-profile egress2
Egress Buffer Profile: egress2
Ports attached:  2/1/4

Port share level: level3-1/16
```

# History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |
| 08.0.60 | This command displays the share port level information for the Ruckus ICX 7150. |

# show qos ingress-buffer-profile

Displays information about ingress buffer profiles.

## Syntax

**show qos ingress-buffer-profile** [ *user-profile-name* | **all** ]

## Parameters

*user-profile-name*

Displays information for the specified ingress buffer profile.

**all**

Displays information for all the ingress buffer profiles configured in the system and a list of their XOFF threshold levels.

## Modes

Global configuration mode

## Examples

The following example displays information for all the ingress buffer profiles configured in the system and their XOFF threshold levels.

```
Device(config)# show qos ingress-buffer-profile all

Ingress Buffer Profile: i1
Ports attached:  1/1/1
Per PG Detail:           XOFF Level:
PG 0                     level1-1/64
PG 1                     level3-1/16
PG 2                     level4-1/9
PG 3                     level5-1/5

Ingress Buffer Profile: ing1
Ports attached:  --
Per PG Detail:           XOFF Level:
PG 0                     level6-1/3
PG 1                     level2-1/32
PG 2                     level2-1/32
PG 3                     level2-1/32
```

## History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was introduced. |

# show qos priority-to-pg

Displays priority-to-priority-group (PG) mapping for priority flow control (PFC).

## Syntax

**show qos priority-to-pg**

## Modes

Global configuration mode

## Usage Guidelines

This command displays priority-to-PG mapping for the following flow control modes:

- PFC
- Symmetrical flow control
- Asymmetrical flow control

## Examples

The following example shows priority-to-PG mapping for PFC.

```
Device(config)# show qos priority-to-pg

QoS Internal Priority 0 mapped to Priority Group 0
QoS Internal Priority 1 mapped to Priority Group 0
QoS Internal Priority 2 mapped to Priority Group 1
QoS Internal Priority 3 mapped to Priority Group 1
QoS Internal Priority 4 mapped to Priority Group 1
QoS Internal Priority 5 mapped to Priority Group 2
QoS Internal Priority 6 mapped to Priority Group 2
QoS Internal Priority 7 mapped to Priority Group 4
```

The following example shows priority-to-PG mapping for 802.3x (Flow-Control). Honor is enabled.

```
Device(config)# show qos priority-to-pg

QoS Internal Priority 0 mapped to Priority Group 0
QoS Internal Priority 1 mapped to Priority Group 0
QoS Internal Priority 2 mapped to Priority Group 1
QoS Internal Priority 3 mapped to Priority Group 1
QoS Internal Priority 4 mapped to Priority Group 1
QoS Internal Priority 5 mapped to Priority Group 2
QoS Internal Priority 6 mapped to Priority Group 2
QoS Internal Priority 7 mapped to Priority Group 4
```

The following example shows priority-to-PG mapping for symmetrical flow control for 802.3x (Flow-Control) in Both mode (Generate and Honor are enabled) or Generate-only mode.

```
Device(config)# symmetrical-flow-control enable
Device(config)# show qos priority-to-pg

QoS Internal Priority 0 mapped to Priority Group 7
QoS Internal Priority 1 mapped to Priority Group 7
QoS Internal Priority 2 mapped to Priority Group 7
QoS Internal Priority 3 mapped to Priority Group 7
QoS Internal Priority 4 mapped to Priority Group 7
QoS Internal Priority 5 mapped to Priority Group 2
QoS Internal Priority 6 mapped to Priority Group 2
QoS Internal Priority 7 mapped to Priority Group 4
```

The following example enables flow control on all priorities and shows the priority-to-PG mapping.

```
Device(config)# symmetrical-flow-control enable all
Device(config)# show qos priority-to-pg

QoS Internal Priority 0 mapped to Priority Group 7
QoS Internal Priority 1 mapped to Priority Group 7
QoS Internal Priority 2 mapped to Priority Group 7
QoS Internal Priority 3 mapped to Priority Group 7
QoS Internal Priority 4 mapped to Priority Group 7
QoS Internal Priority 5 mapped to Priority Group 7
QoS Internal Priority 6 mapped to Priority Group 7
QoS Internal Priority 7 mapped to Priority Group 4
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# show qos scheduler-profile

Displays information about scheduler profiles.

## Syntax

**show qos scheduler-profile** { **all** *user-profile-name*}

## Parameters

**all**

Displays information for all the scheduler profiles configured in the system and a list of all the ports attached to any scheduler profile.

*user-profile-name*

Displays information for the specified scheduler profile only.

## Modes

Global configuration mode

## Usage Guidelines

A scheduler profile must be configured before it can be displayed.

Information can be displayed for a maximum of eight scheduler profiles.

On ICX 7750 and ICX 7450 devices this command also displays information for multicast queue weights.

## Examples

The following example displays information for a scheduler profile named user1.

```
Device(config)# show qos scheduler-profile user1

User Scheduler Profile: user1   Scheduling Option: Weighted round-robin
Ports attached:  1/1/1
Per Queue details:    Bandwidth%
Traffic Class 0             1%
Traffic Class 1             1%
Traffic Class 2            10%
Traffic Class 3            10%
Traffic Class 4            10%
Traffic Class 5            10%
Traffic Class 6            20%
Traffic Class 7            38%
```

The following example displays information for all the scheduler profiles configured in the system.

```
Device(config)# show qos scheduler-profile all

User Scheduler Profile: user1   Scheduling Option: Weighted round-robin
Ports attached:  1/1/1
Per Queue details:    Bandwidth%
Traffic Class 0             1%
Traffic Class 1             1%
Traffic Class 2            10%
Traffic Class 3            10%
Traffic Class 4            10%
Traffic Class 5            10%
Traffic Class 6            20%
Traffic Class 7            38%

User Scheduler Profile: user2   Scheduling Option: Strict scheduling
Ports attached:  --

User Scheduler Profile: user3   Scheduling Option: Mixed-SP-WRR
Ports attached:  --
Per Queue details:    Bandwidth%
Traffic Class 0            15%
Traffic Class 1            15%
Traffic Class 2            15%
Traffic Class 3            15%
Traffic Class 4            15%
Traffic Class 5            25%
Traffic Class 6            sp
Traffic Class 7            sp

User Scheduler Profile: user4   Scheduling Option: Weighted round-robin
Ports attached:  --
Per Queue details:    Bandwidth%
Traffic Class 0             3%
Traffic Class 1             3%
Traffic Class 2             3%
Traffic Class 3             3%
Traffic Class 4             3%
Traffic Class 5             3%
Traffic Class 6             7%
Traffic Class 7            75%
```

The following example displays information, including multicast queue weights, for a scheduler profile named profle1 on ICX 7450 anf ICX 7750 devices.

```
Device(config)# show qos scheduler-profile profile1
User Scheduler Profile: profile1     Scheduling Option: Weighted round-robin
Unicast per Queue details:      Bandwidth%
Traffic Class 0                      8%
Traffic Class 1                      8%
Traffic Class 2                      8%
Traffic Class 3                      8%
Traffic Class 4                      8%
Traffic Class 5                      8%
Traffic Class 6                      8%
Traffic Class 7                     44%
Multicast per Queue details:    Bandwidth%
Traffic Class 0,1                   16%
Traffic Class 2,3,4                 24%
Traffic Class 5                      8%
Traffic Class 6,7                   52%
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |
| 8.0.20 | This command was modified to display information for multicast queue weights on ICX 7450 and ICX 7750 devices. |

# show qos sflow-rate-limit

Displays the CPU rate limit for sFlow.

## Syntax

**show qos sflow-rate-limit**

## Modes

Global configuration mode

## Examples

To view the CPU rate limit for sFlow use the following command.

```
device(config)# show qos sflow-rate-limit
Queue-Num       Rate-Limt       Burst-Size
Queue13           100             5000
device(config)#
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show qos-internal-trunk-queue

Displays the queue-share level of inter-packet-processor (inter-pp) links used to connect master and slave units in ICX 7450 devices.

## Syntax

**show qos-internal-trunk-queue**

## Modes

Global configuration mode

## Examples

The following example displays the queue-share level applied on egress queues of inter-pp links in a system.

```
device(config)#show qos-internal-trunk-queue
Per Queue Details:       Share Level:
Queue 0                  level7-1/2
Queue 1                  level3-1/16
Queue 2                  level3-1/16
Queue 3                  level3-1/16
Queue 4                  level3-1/16
Queue 5                  level3-1/16
Queue 6                  level3-1/16
Queue 7                  level3-1/16
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was introduced. |

# show qos-profiles

Displays information about QoS profiles

## Syntax

**show qos-profiles** { **all** | *name* }

## Parameters

**all**

Displays information for all profiles.

*name*

Displays information for the specified profile.

## Modes

Global configuration mode

## Examples

The following example displays information, including multicast queue weights, for all the queues.

```
device# show qos-profiles all
bandwidth scheduling mechanism: mixed weighted priority with strict priority
Unicast Traffic
Profile qosp7     : Priority7(Highest) Set as strict priority
Profile qosp6     : Priority6          Set as strict priority
Profile qosp5     : Priority5          bandwidth requested  25% calculated  25%
Profile qosp4     : Priority4          bandwidth requested  15% calculated  15%
Profile qosp3     : Priority3          bandwidth requested  15% calculated  15%
Profile qosp2     : Priority2          bandwidth requested  15% calculated  15%
Profile qosp1     : Priority1          bandwidth requested  15% calculated  15%
Profile qosp0     : Priority0(Lowest)  bandwidth requested  15% calculated  15%
Multicast Traffic
Profile qosp7+qosp6                    : Priority7(Highest),6      Set as strict priority
Profile qosp5                          : Priority5                 bandwidth requested  25%
calculated  25%
Profile qosp4+qosp3+qosp2              : Priority4,3,2             bandwidth requested  45%
calculated  45%
Profile qosp1+qosp0                    : Priority1,0(Lowest)       bandwidth requested  30%
calculated  30%
```

## History

| Release version | Command history |
|---|---|
| 08.0.20 | This command was modified to display information for multicast queue weights on ICX 7450 and ICX 7750 devices. |

# show qos-tos

Displays mappings in the DSCP to the forwarding priority portion of the QoS information display.

## Syntax

**show qos-tos**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example displays mappings in the DSCP to forwarding priority portion of the QoS information display. To read this part of the display, select the first part of the DSCP value from the d1 column and select the second part of the DSCP value from the d2 row.

```
device# show qos-tos
DSCP-Priority map: (dscp = d1d2)
     d2|  0   1   2   3   4   5   6   7   8   9
  d1   |
  -----+-----------------------------------------
   0   |  1
   0   1   1   1
   0   0   0   5
   1
   1   |  6   1   1   1   1   1   4
   2   2  2
   2   |  2   2   2   2   2
   3   3  3   3   3
   3   |  3   3   0
   4   4  4   4   4   4   4
   4   |  7
   5   5  5   5   5   5   5   3
   6
   5   |  6   6   6   6   6   6   6
   7   7  7
   6   |  7   7   7   7
```

# show radius servers

Displays the current status of the linked RADIUS servers.

## Syntax

**show radius servers**

## Modes

User EXEC mode

## Command Output

The **show radius servers** command displays the following information:

| Output field | Description |
|---|---|
| Server | The IP address of the RADIUS server. |
| Type | What type of functionality the RADIUS server provides. |
| Opens | The number of times the path to the RADIUS server opens. |
| Closes | The number of times the path to the RADIUS server closes. |
| Timeouts | The number of times the path to the RADIUS server times out. |
| Status | The current status of the path to the RADIUS server. |

## Examples

The following example shows output for the **show radius servers** command.

```
device> show radius servers

-------------------------------------------------------------------------------
Server                      Type      Opens     Closes   Timeouts   Status
-------------------------------------------------------------------------------
10.21.240.60                 any          0          0          0   active
```

# show rate-limit broadcast

Displays the broadcast limit configured on the device.

## Syntax

**show rate-limit broadcast**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example is sample output from the **show rate-limit broadcast** command. The output displays the broadcast limit or broadcast and multicast limit for each port to which it applies.

```
device# show rate-limit broadcast

Broadcast/Multicast Limit Settings:
Port    Limit           Packets/Bytes      Packet Type(s)
4       1245184         Bytes              Broadcast + Multicast Bytes
14      65536           Packets            Broadcast only
23      131072          Packets            Broadcast + Multicast
```

# show rate-limit input

Displays the fixed rate limiting configuration.

## Syntax

**show rate-limit input**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show rate-limit input** command displays the following information.

| Output field | Description |
|---|---|
| Total rate-limited interface count | The total number of ports that are configured for fixed rate limiting. |
| Port | The configured port number. |
| Configured Input Rate | The maximum rate requested for inbound traffic. The rate is measured in kilobits per second (kbps). |
| Actual Input Rate | The actual maximum rate provided by the hardware. The rate is measured in kbps. |

## Examples

The following example is sample output from the **show rate-limit input** command. The command lists the ports on which fixed rate limiting is configured.

```
device# show rate-limit input
Total rate-limited interface count: 5.
    Port       Configured Input Rate    Actual Input Rate
    1/1/1      65000                    65000
    1/1/2      195000                   195000
    1/1/6      1950                     1950
    1/5/2      230432                   230000
    1/5/6      234113                   234000
```

# show rate-limit output-shaping

Displays the configured outbound rate shaper.

## Syntax

**show rate-limit output-shaping**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example is sample output from the **show rate-limit output-shaping** command. The display lists the ports on a device, the configured outbound rate shaper on a port and for a priority for a port.

```
device# show rate-limit output-shaping

Outbound Rate Shaping Limits in Kbps:
Port     PortMax     Prio0     Prio1     Prio2     Prio3     Prio4     Prio5     Prio6     Prio7
1/1/1    -               -         -         -         -         -         -         651
1/1/2    1302            -         -         -         -         -         -         -
1/1/5    651             -         -         -         -         -         -         -
```

# show rate-limit unknown-unicast

Displays the unknown unicast limit for each port region to which it applies.

## Syntax

**show rate-limit unknown-unicast**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example is sample output from the **show rate-limit unknown-unicast** command. The output displays the unknown unicast limit for each port region to which it applies.

```
device# show rate-limit unknown-unicast
Unknown Unicast Limit Settings:
Port Region     Combined Limit     Packets/Bytes
1 - 12          524288             Packets
13 - 24         65536              Bytes
```

# show rear-module

Displays operational status and configuration of the rear module.

## Syntax

**show rear-module**

## Modes

Privileged Exec mode

## Usage Guidelines

The command applies to ICX 7650 devices only.

## Examples

The following example shows an ICX 7650 unit with the rear module operating in default mode (100-Gbps stacking).

```
ICX7650-48P Router# show rear-module
The rear module operates in stacking mode with 100G speed.
The rear module is configured in stacking mode with 100G speed.
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# show relative-utilization

Displays utilization percentages for an uplink.

## Syntax

**show relative-utilization** *num*

## Parameters

*num*

Specifies the utilization list number. The value can range from 1 to 4.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

After you configure an uplink utilization list, you can display the list to observe the percentage of uplink bandwidth that each downlink port used during the most recent 30-second port statistics. The number of packets sent and received between the two ports is listed, as well as the ratio of each downlink port's packets relative to the total number of packets on the uplink.

## Examples

The following is sample output from the **show relative-utilization** command.

```
device# show relative-utilization 1

uplink: ethe 1/1/1
30-sec total uplink packet count = 2996
packet count ratio (%)
1 /2:100 1/ 3:---
```

# show reserved-vlan-map

Displays the assigned VLAN IDs for reserved VLANs.

## Syntax

**show reserved-vlan-map**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

To view the assigned VLAN IDs for reserved VLANs 4091 and 4092, use the **show reserved-vlan-map** command. The reassigned VLAN IDs also display in the output of the **show running-config** and **show config** commands.

## Command Output

The **show reserved-vlan-map** command displays the following information:

| Output field | Description |
|---|---|
| Reserved Purpose | The reason the VLAN is reserved. |
| Default | The default VLAN ID of the reserved VLAN. |
| Re-assign | The VLAN ID to which the reserved VLAN was reassigned. |
| Current | The current VLAN ID for the reserved VLAN. |

**NOTE**
If you reassign a reserved VLAN without saving the configuration and reloading the software, the reassigned VLAN ID will display in the Re-assign column. However, the previously configured or default VLAN ID will display in the Current column until the configuration is saved and the device reloaded.

## Examples

The following is a sample output of the **show reserved-vlan-map** command.

```
device> show reserved-vlan-map
Reserved Purpose     Default   Re-assign   Current
 CPU VLAN              4091       10         33
 All Ports VLAN        4092       10         33
```

# show rmon

Displays the Remote monitoring (RMON) agent status and information about RMON alarms, events, history, logs, and statistics on the interface.

## Syntax

**show rmon** { **alarm** *alarm-number* | **event** *event-number* | **history** *history-index* | **logs** *event-index* | **statistics** [ *number* | *interface-type* | *interface-number* ] }

## Parameters

**alarm**
Specifies to display the RMON alarm table.

*alarm-number*
Specifies the alarm index identification number. Valid values range from 1 through 65535.

**event**
Specifies to display the RMON event table.

*event-number*
Specifies the event index identification number. Valid values range from 1 through 65535.

**history**
Specifies to display the history control data entries for port or interface.

*history-number*
Specifies the history index identification number of the history entry.

**logs**
Specifies to display the RMON logging table where RMON log entries are stored.

*event-index*
Specifies the event index identification number. Valid values range from 1 through 65535.

**statistics**
Specifies to display the RMON Ethernet statistics; and the statistics group that collects statistics on promiscuous traffic across an interface and total traffic into and out of the agent interface. Valid values range from 1 through 65535.

*statistics-number*
Specifies the statistics index identification number of the statistics entry.

*interface-type*
Specifies the ethernet interface or management port.

*interface-number*
Specifies the interface or management port number.

## Modes

Privileged EXEC mode

Global configuration mode

# Command Output

The **show rmon** command displays the following information:

| Output field | Description |
|---|---|
| Rising threshold | The sampling value limit, beyond which the rising alarm is triggered. |
| Falling threshold | The sampling value limit, beyond which the falling alarm is triggered. |
| Octets | The total number of octets of data received on the network. This number includes octets in bad packets. This number does not include framing bits but does include Frame Check Sequence (FCS) octets. |
| Drop events | Indicates an overrun at the port. The port logic could not receive the traffic at full line rate and had to drop some packets as a result. The counter indicates the total number of events in which packets were dropped by the RMON probe due to lack of resources. This number is not necessarily the number of packets dropped, but is the number of times an overrun condition has been detected. |
| Packets | The total number of packets received. This number includes bad packets, broadcast packets, and multicast packets. |
| Broadcast pkts | The total number of good packets received that were directed to the broadcast address. This number does not include multicast packets. |
| Multicast pkts | The total number of good packets received that were directed to a multicast address. This number does not include packets directed to the broadcast address. |
| CRC align errors | The total number of packets received that were from 64 - 1518 octets long, but had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error). The packet length does not include framing bits but does include FCS octets. |
| Undersize pkts | The total number of packets received that were less than 64 octets long and were otherwise well formed. This number does not include framing bits but does include FCS octets. |
| Fragments | The total number of packets received that were less than 64 octets long and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error). It is normal for this counter to increment, since it counts both runts (which are normal occurrences due to collisions) and noise hits. This number does not include framing bits but does include FCS octets. |
| Oversize packets | The total number of packets received that were longer than 1518 octets and were otherwise well formed. This number does not include framing bits but does include FCS octets.<br><br>**NOTE**<br>48GC modules do not support count information on oversized packets and report 0. |
| Jabbers | The total number of packets received that were longer than 1518 octets and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error).<br><br>**NOTE**<br>This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition where any packet exceeds 20 ms. The allowed range to detect jabber is between 20 ms and 150 ms.<br><br>This number does not include framing bits but does include FCS octets.<br><br>**NOTE**<br>48GC modules do not support count information on jabbers and report 0. |
| Collisions | The best estimate of the total number of collisions on this Ethernet segment. |

| Output field | Description |
|---|---|
| 64 octets pkts | The total number of packets received that were 64 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 65 to 127 octets pkts | The total number of packets received that were 65 - 127 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 128 to 255 octets pkts | The total number of packets received that were 128 - 255 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 256 to 511 octets pkts | The total number of packets received that were 256 - 511 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 512 to 1023 octets pkts | The total number of packets received that were 512 - 1023 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 1024 to 1518 octets pkts | The total number of packets received that were 1024 - 1518 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| Event Index | The event index identification number. |
| Log Index | The log index identification number. |
| Log Generated time | The time at which the log is generated. |
| Log Description | Indicates the type of alarm; whether it is a rising or falling alarm. |

# Examples

The following example shows the output of the **show rmon alarm** command.

```
device(config)# show rmon alarm
Alarm 1 is active, owned by monitor
 Monitors etherStatsPkts.13 every 5 seconds
 Taking absolute samples, last value was 675
 Rising threshold is 100, assigned to event 1
 Falling threshold is 0, assigned to event 1
 On startup enable rising or falling alarm

Alarm 2 is active, owned by monitor
 Monitors etherStatsPkts.2 every 5 seconds
 Taking absolute samples, last value was 414
 Rising threshold is 100, assigned to event 3
 Falling threshold is 0, assigned to event 3
 On startup enable rising or falling alarm
```

The following example shows the output of the **show rmon event** command.

```
device(config)# show rmon event
Event 1 is active, owned by monitor
 Description is testing
 Event firing causes log, community
 Batch ID 0, argument <none>
 Last fired at system up time 3 minutes 52 seconds

Event 2 is active, owned by monitor
 Description is logging
 Event firing causes log and trap, community public
 Batch ID 0, argument <none>
 Last fired at system up time 8 minutes 12 seconds
```

The following example shows the output of the **show rmon history** *history-index* command.

```
device(config)# show rmon history 1
History 1 is active, owned by monitor
 Monitors interface mgmt1 (ifIndex 25) every 30 seconds
 25 buckets were granted to store statistics
```

The following example shows the output of the **show rmon logs** command.

```
device(config)# show rmon logs
 Event Index = 1
     Log Index = 1
     Log Generated time = 00:03:52 (23200)
     Log Description  = rising alarm

 Event Index = 2
     Log Index = 1
     Log Generated time = 00:08:12 (49200)
     Log Description  = rising alarm

 Event Index = 3
     Log Index = 1
     Log Generated time = 00:05:12 (31200)
     Log Description  = rising alarm

 Event Index = 4
     Log Index = 1
     Log Generated time = 00:01:32 (9200)
     Log Description  = falling alarm

     Log Index = 2
     Log Generated time = 00:02:52 (17200)
     Log Description  = rising alarm
```

The following example shows the output of the **show rmon logs** *event-index* command.

```
device(config)# show rmon logs 2
Event Index = 2
     Log Index = 1
     Log Generated time = 00:08:12 (49200)
     Log Description  = rising alarm
```

The following example shows the output of the **show rmon statistics** *number* command.

```
device(config)# show rmon statistics 1
Ethernet statistics 1 is active, owned by monitor
 Interface 1/1/1 (ifIndex 1) counters
           Octets                    0
      Drop events                    0             Packets                    0
   Broadcast pkts                    0      Multicast pkts                    0
 CRC align errors                    0      Undersize pkts                    0
    Oversize pkts                    0           Fragments                    0
          Jabbers                    0          Collisions                    0

 Packet size counters
              64                     0           65 to 127                    0
       128 to 255                    0          256 to 511                    0
      512 to 1023                    0        1024 to 1518                    0
```

The following example shows the statistics of the ethernet interface 1/2/1.

```
device(config)# show rmon statistics ethernet 1/2/1
Ethernet statistics 65 is active, owned by monitor
 Interface 1/2/1 (ifIndex 65) counters
           Octets          30170677670
      Drop events                    0            Packets             72281139
   Broadcast pkts                    0      Multicast pkts            66309417
 CRC align errors                    0      Undersize pkts                   0
    Oversize pkts                    0           Fragments                   0
          Jabbers                    0          Collisions                   0

 Packet size counters
               64                    0          65 to 127             10703415
      128 to 255             19353559          256 to 511             18658554
      512 to 1023            17980963        1024 to 1518              5584648
```

# History

| Release version | Command history |
|---|---|
| 08.0.20 | The **logs** keyword was introduced. |

# show rmon statistics

Displays a textual summary of the Remote Monitoring (RMON) statistics for all ports.

## Syntax

**show rmon statistics** [ *ifIndex* | **ethernet** *stack-id/slot/port* | **management** *number* ]

## Parameters

*ifIndex*
Specifies the ifIndex number, in decimal.

**ethernet** *stack-id/slot/port*
Displays the RMON statistics for a specific Ethernet interface.

**management** *number*
Displays the RMON statistics table for the management interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

Counts of multicast and broadcast packets, total packets sent, undersized and oversized packets, CRC alignment errors, jabbers, collisions, fragments, and dropped events are collected for each port on a Ruckus ICX Layer 2 switch or Layer 3 switch. The statistics group collects statistics on promiscuous traffic across an interface. The interface group collects statistics on total traffic in and out the agent interface. No configuration is required to collect statistics for the Layer 2 switch or Layer 3 switch. This activity is by default automatically activated at system startup.

Though 48GC modules receive oversized packets and jabbers, they do not support counts of oversized packets and jabbers, and the output of the **show rmon statistics** command reports 0 for both of these counters.

## Command Output

The **show rmon statistics** command displays the following information.

| Output field | Description |
|---|---|
| Octets | The total number of octets of data received on the network. This number includes octets in bad packets. This number does not include framing bits but does include Frame Check Sequence (FCS) octets. |
| Drop events | The total number of times an overrun condition has been detected at the port. The port logic could not receive the traffic at full line rate and had to drop some packets as a result. The counter indicates the total number of events in which packets were dropped by the |

| Output field | Description |
|---|---|
| | RMON probe due to lack of resources. This number is not necessarily the number of packets dropped, but it is the number of times an overrun condition has been detected. |
| Packets | The total number of packets received. This number includes bad packets, broadcast packets, and multicast packets. |
| Broadcast pkts | The total number of good packets received that were directed to the broadcast address. This number does not include multicast packets. |
| Multicast pkts | The total number of good packets received that were directed to a multicast address. This number does not include packets directed to the broadcast address. |
| CRC align errors | The total number of packets received that were from 64 to 1518 octets long, but had either a bad FCS with an integral number of octets (FCS error) or a bad FCS with a nonintegral number of octets (alignment error). The packet length does not include framing bits but does include FCS octets. |
| Fragments | The total number of packets received that were less than 64 octets long and had either a bad FCS with an integral number of octets (FCS error) or a bad FCS with a nonintegral number of octets (alignment error). It is normal for this counter to increment, because it counts both runts (which are normal occurrences due to collisions) and noise hits. This number does not include framing bits but does include FCS octets. |
| Undersize pkts | The total number of packets received that were less than 64 octets long but were otherwise well formed. This number does not include framing bits but does include FCS octets. |
| Oversize packets | The total number of packets received that were longer than 1518 octets but were otherwise well formed. This number does not include framing bits but does include FCS octets.<br><br>**NOTE**<br>48GC modules do not support counts of oversized packets and report 0. |
| Jabbers | The total number of packets received that were longer than 1518 octets and had either a bad FCS with an integral number of octets (FCS error) or a bad FCS with a nonintegral number of octets (alignment error). This number does not include framing bits but does include FCS octets.<br><br>**NOTE**<br>This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition in which any packet exceeds 20 ms. The allowed range to detect jabber is between 20 ms and 150 ms.<br><br>**NOTE**<br>48GC modules do not support counts of jabbers and report 0. |
| Collisions | The best estimate of the total number of collisions on this Ethernet segment. |
| 64 octets pkts | The total number of packets received that were 64 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 65 to 127 octets Pkts | The total number of packets received that were from 65 to 127 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 128 to 255 octets Pkts | The total number of packets received that were from 128 to 255 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 256 to 511 octets Pkts | The total number of packets received that were from 256 to 511 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |
| 512 to 1023 octets Pkts | The total number of packets received that were from 512 to 1023 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |

| Output field | Description |
|---|---|
| 1024 to 1518 octets pkts | The total number of packets received that were from 1024 to 1518 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets. |

# Examples

The following is sample output from the **show rmon statistics** command.

```
device# show rmon statistics

Ethernet statistics 1 is active, owned by monitor
 Interface 1/1/1 (ifIndex 1) counters
           Octets                    0
      Drop events                    0             Packets                    0
   Broadcast pkts                    0      Multicast pkts                    0
 CRC align errors                    0      Undersize pkts                    0
    Oversize pkts                    0           Fragments                    0
          Jabbers                    0          Collisions                    0

 Packet size counters
              64                    0           65 to 127                    0
       128 to 255                   0          256 to 511                    0
      512 to 1023                   0       1024 to 10200                    0

Ethernet statistics 2 is active, owned by monitor
 Interface 1/1/2 (ifIndex 2) counters
           Octets                    0
      Drop events                    0             Packets                    0
   Broadcast pkts                    0      Multicast pkts                    0
 CRC align errors                    0      Undersize pkts                    0
    Oversize pkts                    0           Fragments                    0
          Jabbers                    0          Collisions                    0

 Packet size counters
              64                    0           65 to 127                    0
       128 to 255                   0          256 to 511                    0
      512 to 1023                   0       1024 to 10200                    0
```

The following is sample output from the **show rmon statistics** command for ifIndex 9.

```
device# show rmon statistics 9
Ethernet statistics 9 is active, owned by monitor
 Interface 1/1/6 (ifIndex 9) counters
           Octets                    0
      Drop events                    0             Packets                    0
   Broadcast pkts                    0      Multicast pkts                    0
 CRC align errors                    0      Undersize pkts                    0
    Oversize pkts                    0           Fragments                    0
          Jabbers                    0          Collisions                    0

 Packet size counters
              64                    0           65 to 127                    0
       128 to 255                   0          256 to 511                    0
      512 to 1023                   0       1024 to 10200                    0
```

# show rspan-vlan

Displays information about Remote Switched Port Analyzer (RSPAN) VLANs.

## Syntax

**show rspan-vlan**

## Modes

User EXEC mode

## Examples

The following example is sample output for the **show rspan-vlan** command.

```
device# show rspan-vlan

RSPAN details:
VLAN: 20
RSPAN destination port: ethe 1/1/27
RSPAN ingress monitor source port(s): ethe 1/1/43
RSPAN egress monitor source port(s): ethe 1/1/43
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# show running ikev2

Displays current Internet Key Exchange version 2 (IKEv2) configuration information.

## Syntax

**show running ikev2**

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

Use this command to display the IKEv2 configuration that is currently active on the device.

## Examples

The following example displays the current IKEv2 configuration.

```
device# show running ikev2
!
ikev2 proposal ikev2_propA
!
ikev2 auth-proposal ikev2_auth_propA
pre-shared-key 2 $MlVzZCFAblp8OA==
!
ikev2 policy ikev2_policyA
proposal ikev2_propA
match address-local 10.100.100.1 255.255.255.255
!
ikev2 profile ikev2_profA
authentication ikev2_auth_propA
 local-identifier address 10.1.1.1
remote-identifier address 10.4.4.4
match-identity local address 10.1.1.1
match-identity remote address 10.4.4.4
!
!
ipsec proposal ipsec_propA
!
ipsec profile ipsec_profA
proposal ipsec_propA
ike-profile ikev2_profA
!
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show running interface

Displays information about the interface.

## Syntax

**show running interface** [ **ethernet** *stack/slot/port* [ **to ethernet** *stack/slot/port* ] | **loopback** *loopback-number* | **management** *por-id* | **tunnel** *tunnel-id* | **ve** *ve-number*]

## Parameters

**ethernet** *stack/slot/port*

Specifies the configuration on a physical interface. On standalone devices specify the interface ID in the format slot/port-id; on stacked devices you must also specify the stack ID, in the format stack-id/slot/port-id.

**to**

Specifies information for a range of physical interfaces.

**loopback** *loopback-number*

Specifies information for a loopback interface.

**management** *port-id*

Specifiesinformation for a management port.

**tunnel** *tunnel-id*

Specifies information for a tunnel interface.

**ve** *ve-number*

Specifies information for a virtual interface.

## Modes

Privileged EXEC mode

## Examples

The following example displays output from the **show running interface** command, showing that ACLs 10 and f10 are applied to interface 1/1/9 to control neighbor access.

```
Device#show running interface ethernet 1/1/9
interface ethernet 1/1/9
 ip address 15.1.1.5 255.255.255.0
 ip pim-sparse
 ip pim neighbor-filter 10
 ip ospf area 0
 ipv6 address 201::1/64
 ipv6 ospf area 0
 ipv6 pim-sparse
 ipv6 pim neighbor-filter f10
```

# History

| Release version | Command history |
|---|---|
| 8.0.20a | This command was modified to display neighbor filter information. |

# show running-config

Displays the current running configuration.

## Syntax

**show running-config** [ **interface** [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ] | **loopback** *loopback-port-num* | **management** *mgmt-port-num* | **tunnel** *tunnel-port-num* | **ve** *ve-port-num* ] ]

**show running-config** [ **interface** [ **lag** *lag-d* [ **to** *lag-id* | [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ] [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ]... ]

**show running-config** [ **vlan** *vlan-id* ]

**show running-config** [ **vrf** ]

## Parameters

**interface**
Displays the running configuration for the specified interface type.

**ethernet** *unit/slot/port*
Displays the running configuration on the specified Ethernet interface.

**to** *unit/slot/port*
Specifies the range of the Ethernet interface for which to display the running configuration.

**lag** *lag-id*
Specifies the LAG virtual interface.

**to** *lag-id*
Specifies a range of LAG virtual interface IDs.

**loopback** *loopback-port-num*
Displays the running configuration information for the specified loopback interface.

**management** *mgmt-port-num*
Displays the running configuration information for the specified management interface.

**tunnel** *tunnel-port-num*
Displays the running configuration information for the specified tunnel interface.

**ve** *ve-port-num*
Displays the running configuration information for the specified VE port.

**vlan** *vlan-id*
Specifies that web management should be enabled on the clients of the specified VLAN.

**vrf**
Displays the VRF-Lite running configuration.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

Use this command to display the configuration that is currently active on the local switch but which is not saved persistently.

## Examples

The following example displays sample output of the **show running-config vlan** command.

```
device(config)# show running-config vlan 100
vlan 502 by port
 tagged lag 1 ethe 1/2/5
 router-interface ve 502
```

The following example displays sample output of the **show running-config** command.

```
device(config)# show running-config
Current configuration:
!
ver 08.0.20a
!
stack unit 1
  module 1 icx6610-24-port-management-module
  module 2 icx6610-qsfp-10-port-160g-module
  module 3 icx6610-8-port-10g-dual-mode-module
  stack-trunk 1/2/1 to 1/2/2
  stack-trunk 1/2/6 to 1/2/7
!
!
!
lag red dynamic id 1
!
!
vlan 1 name DEFAULT-VLAN by port
!
vlan 2 by port
!
vlan 10 by port
 router-interface ve 10
 multicast port-version 3 ethe 1/1/3
 multicast6 fast-leave-v1
!
vlan 20 by port
 untagged ethe 1/1/5
 multicast port-version 3 ethe 1/1/5
!
vlan 150 by port
!
!
!
!
openflow enable ofv130
!
system-max pim6-hw-mcache 726
!
vrf blue
 rd 10.1.0.1:10
exit-vrf
!
vrf my_vrf
exit-vrf
!
vrf 3
exit-vrf
!
vrf vrf2
exit-vrf
!
vrf mroute
exit-vrf
!
vrf config'
exit-vrf
!
vrf config
exit-vrf
!
(output truncated)
```

The following example is sample output from the **show running-config** command for a switch, including dynamically obtained DHCP options.

```
device> show running-config

Current configuration:
!
ver 08.0.61b1T211
!
stack unit 1
  module 1 icx7250-24-port-management-module
  module 2 icx7250-sfp-plus-8port-80g-module
!
!
!
vlan 1 name DEFAULT-VLAN by port
!
!
!
hostname TestHostName dynamic
ip address 10.10.10.2 255.255.255.0 dynamic
ip dns domain-list ManualDomain.com
ip dns domain-list testStaticDomain.com
ip dns domain-list testDomain.com dynamic
ip dns server-address 20.20.20.8 20.20.20.9 20.20.20.5 10.10.10.5(dynamic)
ip default-gateway 10.10.10.1 dynamic
!
!
!
interface ethernet 1/1/21
 disable
!
interface ethernet 1/2/2
 speed-duplex 1000-full
!
interface ethernet 1/2/4
 speed-duplex 1000-full
!
interface ethernet 1/2/5
 speed-duplex 1000-full
!
interface ethernet 1/2/6
 speed-duplex 1000-full
!
interface ethernet 1/2/7
 speed-duplex 1000-full
!
interface ethernet 1/2/8
 speed-duplex 1000-full
!
!
!
lldp run
!
!
end
```

The following example is sample output from the **show running-config** command for a router, including dynamically obtained DHCP options.

```
device> show running-config

Current configuration:
!
ver 08.0.61b1T213
!
stack unit 1
  module 1 icx7250-24-port-management-module
  module 2 icx7250-sfp-plus-8port-80g-module
!
!
vlan 1 name DEFAULT-VLAN by port
!
!
!
hostname TestHostName dynamic
ip dns domain-list ManualDomain.com
ip dns domain-list testDomain.com dynamic
ip dns domain-list testStaticDomain.com
ip dns server-address 20.20.20.8 20.20.20.9 10.10.10.5(dynamic) 20.20.20.5
ip route 0.0.0.0/0 10.10.10.1 distance 254 dynamic
!
!
interface ethernet 1/1/7
 ip address 10.10.10.2 255.255.255.0 dynamic
!
interface ethernet 1/1/21
 disable
!
interface ethernet 1/2/2
 speed-duplex 1000-full
!
interface ethernet 1/2/4
 speed-duplex 1000-full
!
interface ethernet 1/2/5
 speed-duplex 1000-full
!
interface ethernet 1/2/6
 speed-duplex 1000-full
!
interface ethernet 1/2/7
 speed-duplex 1000-full
!
interface ethernet 1/2/8
 speed-duplex 1000-full
!
!
lldp run
!
!
end
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. This command was modified to include information about dynamically obtained DHCP options. |

# show running-config interface ethernet

Displays the status of a specific Ethernet interface.

## Syntax

**show running-config interface ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ]

## Parameters

*unit / slot / port*

Stack ID number, slot number, and port number for an existing Ethernet interface.

**to** *unit/slot/port*

Specifies a range of Ethernet interfaces.

**lag** *lag-id*

Specifies the LAG virtual interface.

**to** *lag-id*

Specifies a range of LAG virtual interface IDs.

## Modes

Privileged EXEC mode

## Examples

This example displays the running configuration for an Ethernet interface including the configured bandwidth.

```
device# show running-config interface ethernet 1/1/9
interface ethernet 1/1/9
 bandwidth 2000
 ip address 10.1.1.5 10.255.255.0
 ip pim
 ip ospf area 0
 ipv6 address 201::1/64
 ipv6 ospf area 0
 ipv6 pim-sparse
 ipv6 pim dr-priority 50
 ipv6 pim border
 ipv6 mld version 2
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was modified to include configured bandwidth status. |
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# show running-config interface tunnel

Displays the status of a specific tunnel interface.

## Syntax

**show running-config interface tunnel** { *tunnel-number* }

## Parameters

*tunnel-number*

Specifies the tunnel number.

## Modes

Privileged EXEC mode

## Examples

This example displays the running configuration for a tunnel interface, including the configured bandwidth.

```
device# show running-config interface tunnel 2

interface tunnel 2
 tunnel mode gre ip
 tunnel source 10.0.0.1
 tunnel destination 10.10.0.1
 ip address 10.0.0.1/24
 bandwidth 2000
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was modified to include configured bandwidth status. |

# show running-config interface ve

Displays the status of a specific Virtual Ethernet (VE) interface.

## Syntax

**show running-config interface ve** { *vlan_id* }

## Parameters

*vlan_id*

Specifies the configured corresponding VLAN interface.

## Modes

Privileged EXEC mode

## Examples

This example displays the running configuration for a VE interface, including the configured bandwidth.

```
device# show running-config interface ve 20
interface ve 20
 ip address 10.21.21.22 10.255.255.0
 ip pim-sparse
 ip ospf area 0
 bandwidth 2000
 ipv6 address 2000::2/64
 ipv6 ospf area 0
```

## History

| Release version | Command history |
|---|---|
| 8.0.30 | This command was modified to include configured bandwidth status. |

# show running-config vlan

Displays information about all VLANs or a specified VLAN.

## Syntax

**show running-config vlan** [ *vlan-id* ]

## Parameters

*vlan-id*

Specifies the VLAN ID.

## Modes

User EXEC mode

## Examples

The following example is sample output for the **show running-config vlan** command.

```
device# show running-config vlan

vlan1 name DEFAULT-VLAN by port
spanning-tree802-1w
!
rspan-vlan 20
tagged ethe 1/1/27 ethe 1/1/45
rspan destination ethe 1/1/27
rspan source monitor-in ethe 1/1/43
rspan source monitor-out ethe 1/1/43
!
vlan30 by port
tagged ethe 1/1/43
!
!
```

## History

| Release version | Command history |
|---|---|
| 8.0.80 | This command was modified to include information about configured Remote Switched Port Analyzer (RSPAN) VLANs. |

# show scheduler-profile

Displays the user-configurable scheduler profile configuration.

## Syntax

**show scheduler-profile** { *user-profile-name* | **all** }

## Parameters

*user-profile-name*

Displays the configured scheduler profile for the specified profile.

**all**

Displays all scheduler profiles in the runtime configuration for the system.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following example is sample output from the **show scheduler-profile all** command.

```
device(config)# show scheduler-profile all

User Profile: profile1  Scheduling Option: Mixed-SP-WRR
Per Queue details:    Bandwidth%
Traffic Class 0       15%
Traffic Class 1       15%
Traffic Class 2       15%
Traffic Class 3       15%
Traffic Class 4       15%
Traffic Class 5       25%
Traffic Class 6       sp
Traffic Class 7       sp
User Profile: profile2  Scheduling Option: Weighted round-robin
Per Queue details:    Bandwidth%
Traffic Class 0       3%
Traffic Class 1       3%
Traffic Class 2       3%
Traffic Class 3       3%
Traffic Class 4       3%
Traffic Class 5       3%
Traffic Class 6       7%
Traffic Class 7       75%
```

# show sflow

Displays the sFlow configuration and statistics.

## Syntax

**show sflow**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

You can display the rates that you configured for the default sampling rate, module rates, and all sFlow-enabled ports. You can view the agent IP address and several other details.

## Command Output

The **show sflow** command displays the following information.

| Output field | Description |
|---|---|
| sFlow version | The version of sFlow enabled on the device, which can be 2 or 5. |
| sFlow services | The feature state, which can be enabled or disabled. |
| sFlow agent IP address | The IP address that sFlow is using in the agent_address field of packets sent to the collectors. |
| Collector | The collector information. The following information is displayed for each collector:<br>• IP address<br>• UDP port<br>If more than one collector is configured, the line above the collectors indicates how many have been configured. |
| Configured UDP source | The UDP source port used to send data to the collector. |
| Polling interval | The polling interval of the port counter. |
| Configured default sampling rate | The configured global sampling rate. If you change the global sampling rate, the value you enter is shown here. The actual rate calculated by the software based on the value you enter is listed on the next line, "Actual default sampling rate." |
| Actual default sampling rate | The actual default sampling rate. |
| The maximum sFlow sample size | The maximum size of a flow sample sent to the sFlow collector. |
| exporting cpu-traffic | Indicates whether the sFlow agent is configured to export data destined to the CPU (for example, Telnet sessions) to the sFlow collector:<br>• enabled<br>• disabled |

| Output field | Description |
|---|---|
| exporting cpu-traffic sample rate | The sampling rate for CPU-directed data, which is the average ratio of the number of incoming packets on an sFlow-enabled port, to the number of flow samples taken from those packets. |
| exporting system-info | Indicates whether the sFlow agent is configured to export information about CPU and memory usage to the sFlow collector:<br>• enabled<br>• disabled |
| exporting system-info polling interval | Specifies the interval, in seconds, at which sFlow data is sent to the sFlow collector. |
| UDP packets exported | The number of sFlow export packets the device has sent.<br>**NOTE**<br>Each UDP packet can contain multiple samples. |
| sFlow samples collected | The number of sampled packets that have been sent to the collectors. |
| sFlow ports | The ports on which you enabled sFlow. |
| Module Sampling Rates | The configured and actual sampling rates for each module. If a module does not have any sFlow-enabled ports, the rates are listed as 0. |
| Port Sampling Rates | The configured and actual sampling rates for each sFlow-enabled port. The subsampling factor indicates how many times the sampling rate of the port's module is multiplied to achieve the port's sampling rate. Because of the way in which the actual sampling rates are computed, the subsampling factors are always whole numbers. |

# Examples

The following is sample output from the **show sflow** command.

```
device# show sflow

sFlow version: 5
sFlow services are enabled.

sFlow agent IP address: 10.1.1.1
3 collector destinations configured:
Collector IP 10.2.2.2, UDP 6343
Collector IP 10.3.3.3, UDP 6343
Collector IP 10.4.4.4, UDP 6343
Configured UDP source port: 9999
Polling interval is 30 seconds.
Configured default sampling rate: 1 per 566 packe
Actual default sampling rate: 1 per 566 packets.
The maximum sFlow sample size: 1200.
Sample mode: All packets including dropped packet
exporting cpu-traffic is enabled.
exporting cpu-traffic sample rate: 18.
exporting system-info is enabled
exporting system-info polling interval: 30 second
22 UDP packets exported
0 sFlow flow samples collected.
sFlow ports: ethe 1/1/1 to 1/1/2
Module Sampling Rates
---------------------
U1:M1 configured rate=300, actual rate=300
Port Sampling Rates
-------------------
Port=1/1/1, configured rate=300, actual rate=300
Port=1/1/2, configured rate=400, actual rate=400
```

# show snmp

Displays various SNMP statistics.

## Syntax

**show snmp** [ **engineid** | **group** | **server** | **user** ]

## Parameters

**engineid**
Displays local and remote SNMP engine IDs.

**group**
Displays SNMP groups.

**server**
Displays SNMP server status and trap information

**user**
Displays SNMPv3 users details.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show snmp engineid** command displays the following information:

| Output field | Description |
|---|---|
| Local SNMP Engine ID | The engine ID that identifies the source or destination of the packet. |
| Engine Boots | The number of times that the SNMP engine reinitialized itself with the same ID. If the engine ID is modified, the boot count is reset to 0. |
| Engine time | The current time with the SNMP agent. |

The **show snmp group** command displays the following information:

| Output field | Description |
|---|---|
| groupname | The SNMP group name configured using the **snmp-server group** command. |
| Security model | Indicates which version of SNMP is used for authentication. SNMP version 3 uses a User-Based Security model (RFC 2574) for authentication and privacy services. SNMP version 1 and version 2 use community strings to authenticate SNMP access to management modules. This method can still be used for authentication. |
| Security level | • none - If the security model shows v1 or v2, then security level is blank. User names are not used to authenticate users; community strings are used instead. |

| Output field | Description |
|---|---|
| | • noauthNoPriv - If the security model shows v3 and user authentication is by user name only.<br>• authNoPriv - If the security model shows v3 and user authentication is by user name and the MD5 or SHA algorithm. |

# Examples

The following example displays output of the **show snmp engineid** command.

```
device# show snmp engineid
Local SNMP Engine ID: 800007c703748ef88315c0
Engine Boots: 24
Engine time: 1586246
```

The following example displays output of the **show snmp group** command.

```
device# show snmp group
groupname = ln
security model = v3
security level = authNoPriv
ACL id = 1
readview = r
writeview = exit
notifyview = n

groupname = d3
security model = v3
security level = authNoPriv
ACL id = 3
readview = all
writeview = all
notifyview = all

groupname = d4
security model = v3
security level = authNoPriv
ACL id = 3
readview = <none>
writeview = <none>
notifyview = 3
```

The following example displays output of the **show snmp server** command.

```
device# show snmp server
       Status: Enabled

      Contact: XYZ
     Location: CopyCenter

Max Ifindex per module: 64

Traps
                   Cold start: Enable
                      Link up: Enable
                    Link down: Enable
               Authentication: Enable
         Power supply failure: Enable
                  Fan failure: Enable
              Fan speed change: Enable
               Module inserted: Enable
                Module removed: Enable
Redundant module state change: Enable
          Temperature warning: Enable
                 STP new root: Enable
          STP topology change: Enable
             MAC notification: Enable
        MAC-AUTH notification: Enable
                         VSRP: Enable
                          MRP: Enable
                         UDLD: Enable
                          VRF: Enable
                    link-oam: Enable
                         cfm: Enable
                     nlp-phy: Enable

 Total Trap-Receiver Entries: 0
```

The following example displays output of the **show snmp user** command.

```
device# show snmp user
username = bob
ACL id = 2
group = admin
security model = v3
group ACL id = 0
authtype = md5
authkey = 3aca18d90b8d172760e2dd2e8f59b7fe
privtype = des, privkey = 1088359afb3701730173a6332d406eec
engine ID= 800007c70300e052ab0000
```

# show span

Displays the Spanning Tree Protocol (STP) information for the device.

## Syntax

**show span** [ *number* | **designated-protect** | **fast-uplink-span** | **pvst-mode** | **root-protect** ]

**show span** [ **detail** [ *number* | **vlan** *vlan-id* [ **ethernet** *stackid/slot/port* | **lag** *lag-id* ] ] ]

**show span** [ **vlan** *vlan-id* [ **ethernet** *stackid/slot/port* | **fast-uplink-span** | **lag** *lag-id* ] ]

## Parameters

*number*
Displays only the entries after the specified number.

**designated-protect**
Displays the designated forwarding state disabled.

**fast-uplink-span**
Displays the status of ports with Fast Uplink Span enabled.

**pvst-mode**
Displays STP information for the device Per VLAN Spanning Tree Plus (PVST+) compatibility configuration.

**root-protect**
Displays the STP root guard state.

**detail**
Displays the detailed STP information for a port.

**vlan** *vlan-id*
Displays the STP information for a VLAN.

**ethernet** *stackid/slot/port*
Displays STP information for an Ethernet port.

**lag** *lag-id*
Specifies STP information for a LAG virtual interface.

## Modes

The command is supported on all command modes.

## Command Output

The **show span** command displays the following information:

| Output field | Description |
|---|---|
| VLAN ID | The port-based VLAN that contains this spanning tree (instance of STP). VLAN 1 is the default VLAN. If you have not configured port-based VLANs on this device, all STP information is for VLAN 1. |

| Output field | Description |
|---|---|
| Root ID | The ID assigned by STP to the root bridge for this spanning tree. |
| Root Cost | The cumulative cost from this bridge to the root bridge. If this device is the root bridge, then the root cost is 0. |
| Root Port | The port on this device that connects to the root bridge. If this device is the root bridge, then the value is "Root" instead of a port number. |
| Priority Hex | This device or VLAN STP priority. The value is shown in hexadecimal format. |
| Max age sec | The number of seconds this device or VLAN waits for a configuration BPDU from the root bridge before deciding the root has become unavailable and performing a reconvergence. |
| Hello sec | The interval between each configuration BPDU sent by the root bridge. |
| Hold sec | The minimum number of seconds that must elapse between transmissions of consecutive Configuration BPDUs on a port. |
| Fwd dly sec | The number of seconds this device or VLAN waits following a topology change and consequent reconvergence. |
| Last Chang sec | The number of seconds since the last time a topology change occurred. |
| Chg cnt | The number of times the topology has changed since this device was reloaded. |
| Bridge Address | The STP address of this device or VLAN. |
| Port Num | The port number. |
| Priority Hex | The port STP priority, in hexadecimal format. |
| Path Cost | The port STP path cost. |
| State | The port STP state. The state can be one of the following:<br>• BLOCKING: STP has blocked Layer 2 traffic on this port to prevent a loop. The device or VLAN can reach the root bridge using another port, whose state is FORWARDING. When a port is in this state, the port does not transmit or receive user frames, but the port does continue to receive STP BPDUs.<br>• DISABLED: The port is not participating in STP. This can occur when the port is disconnected or STP is disabled on the port.<br>• FORWARDING: STP is allowing the port to send and receive frames.<br>• LISTENING: STP is responding to a topology change and this port is listening for a BPDU from neighboring bridges in order to determine the new topology. No user frames are transmitted or received during this state.<br>• LEARNING: The port has passed through the LISTENING state and will change to the FORWARDING state depending on the results of STP reconvergence. The port does not transmit or receive user frames during this state. However, the device can learn the MAC addresses of frames that the port receives during this state and make corresponding entries in the MAC table.<br>• DESIGNATED INCONSISTENT: This shows as DESI-INCONS in the output. You can disallow the designated forwarding state on a port in STP 802.1D or 802.1W with the **spanning-tree designated-protect** command. If STP tries to put this port into the designated forwarding role, the device puts this port into a designated inconsistent STP state. This is effectively equivalent to the listening state in STP in which a port cannot transfer any user traffic. When STP no longer marks this port as a designated port, the device automatically removes the port from the designated inconsistent state. |
| Fwd Trans | The number of times STP has changed the state of this port between BLOCKING and FORWARDING. |
| Design Cost | The cost to the root bridge as advertised by the designated bridge that is connected to this port. If the designated bridge is the root bridge itself, then the cost is 0. The identity of the designated bridge is shown in the Designated Bridge field. |
| Designated Root | The root bridge as recognized on this port. The value is the same as the root bridge ID listed in the Root ID field. |

| Output field | Description |
|---|---|
| Designated Bridge | The designated bridge to which this port is connected. The designated bridge is the device that connects the network segment on the port to the root bridge. |

The **show span detail** command displays the following information:

| Output field | Description |
|---|---|
| Active Spanning Tree protocol | The VLAN that contains the listed ports and the active Spanning Tree Protocol.The STP type can be one of the following:<br>• MULTIPLE SPANNNG TREE (MSTP)<br>• GLOBAL SINGLE SPANNING TREE (SSTP) |
| Bridge identifier | The STP identity of this device. |
| Active global timers | The global STP timers that are currently active, and their current values. The following timers can be listed:<br>• Hello: The interval between Hello packets. This timer applies only to the root bridge.<br>• Topology Change (TC): The amount of time during which the topology change flag in Hello packets will be marked, indicating a topology change. This timer applies only to the root bridge.<br>• Topology Change Notification (TCN): The interval between Topology Change Notification packets sent by a non-root bridge toward the root bridge. This timer applies only to non-root bridges. |
| Active Timers | The current values for the following timers, if active:<br>• Message age: The number of seconds this port has been waiting for a Hello message from the root bridge.<br>• Forward delay: The number of seconds that have passed since the last topology change and consequent reconvergence.<br>• Hold time: The number of seconds that have elapsed since transmission of the last Configuration BPDU. |
| BPDUs Sent and Received | The number of BPDUs sent and received on this port since the software was reloaded. |

# Examples

The following example shows the STP information.

```
device# show span
VLAN 1 BPDU cam_index is 3 and the Master DMA Are(HEX)
STP instance owned by VLAN 1
Global STP (IEEE 802.1D) Parameters:
VLAN      Root              Root    Root    Prio    Max    He-    Ho-    Fwd     Last     Chg  Bridge
ID        ID                Cost    Port    rity    Age    llo    ld     dly     Chang    cnt  Address
                                            Hex     sec    sec    sec    sec     sec
1         800000e0804d4a00  0        Root    8000   20     2      1      15       689      1
00e0804d4a00
Port STP Parameters:
Port          Prio     Path     State        Fwd     Design    Designated        Designated
Num           rity     Cost                  Trans   Cost      Root              Bridge
              Hex
1             80       19       FORWARDING   1       0         800000e0804d4a00  800000e0804d4a00
2             80       0        DISABLED     0       0         0000000000000000  0000000000000000
3             80       0        DISABLED     0       0         0000000000000000  0000000000000000
4             80       0        DISABLED     0       0         0000000000000000  0000000000000000
5             80       19       FORWARDING   1       0         800000e0804d4a00  800000e0804d4a00
6             80       19       BLOCKING     0       0         800000e0804d4a00  800000e0804d4a00
7             80       0        DISABLED     0       0         0000000000000000  0000000000000000
<lines for remaining ports excluded for brevity>
```

The following example shows the detailed STP information.

```
device# show span detail
========================================================================
VLAN 1 - MULTIPLE SPANNING TREE (MSTP) ACTIVE
========================================================================
Bridge identifier - 0x800000e0804d4a00
Active global timers - Hello: 0
Port 1/1/1 is FORWARDING
Port - Path cost: 19, Priority: 128, Root: 0x800000e052a9bb00
Designated - Bridge: 0x800000e052a9bb00, Interface: 1, Path cost: 0
Active Timers - None
BPDUs - Sent: 11, Received: 0
Port 1/1/2 is DISABLED
Port 1/1/3 is DISABLED
Port 1/1/4 is DISABLED <lines for remaining ports excluded for brevity>
```

The following example displays STP information for an individual port in specific a VLAN.

```
device# show span detail vlan 1 ethernet 1/1/1
Port 1/1/1 is FORWARDING
Port - Path cost: 19, Priority: 128, Root: 0x800000e052a9bb00
Designated - Bridge: 0x800000e052a9bb00, Interface: 7, Path cost: 0
Active Timers - None
BPDUs - Sent: 29, Received: 0
```

The following example displays STP information in a VLAN.

```
device# show span vlan 100
STP instance owned by VLAN 100

Global STP (IEEE 802.1D) Parameters:

VLAN Root             Root Root      Prio Max He- Ho- Fwd Last    Chg Bridge
 ID   ID              Cost Port      rity Age llo ld  dly Chang   cnt Address

                                     Hex  sec sec sec sec sec
 100 8000cc4e24b46fcc 0    Root      8000 20  2   1   15  11      1   cc4e24b46fcc

Port STP Parameters:

Port      Prio Path  State       Fwd    Design  Designated       Designated
Num       rity Cost              Trans  Cost    Root             Bridge
          Hex
1/1/1     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

1/1/2     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc
1/1/3     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

1/1/4     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

1/1/5     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

1/1/6     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

1/1/7     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

1/1/8     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

lg1       80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc

lg256     80   4     FORWARDING  1      0       8000cc4e24b46fcc 8000cc4e24b46fcc
```

The following example shows the port path costs after the 802.1D 2004 path cost method is globally configured. The Ethernet 1/1/5 and 1/1/6 port speeds are 1 Gbps.

```
device# show span vlan 100

STP instance owned by VLAN 100

Global STP (IEEE 802.1D) Parameters:

VLAN Root             Root Root      Prio Max He- Ho- Fwd Last    Chg Bridge
 ID   ID              Cost Port      rity Age llo ld  dly Chang   cnt Address
                                     Hex  sec sec sec sec sec
 100 8000cc4e246eb200 0    Root      8000 20  2   1   15  12739   2   cc4e246eb200

Port STP Parameters:

Port      Prio Path      State       Fwd    Design  Designated       Designated
Num       rity Cost                  Trans  Cost    Root             Bridge
          Hex
1/1/5     80   20000     FORWARDING  1      0       8000cc4e246eb200 8000cc4e246eb200
1/1/6     80   20000     FORWARDING  1      0       8000cc4e246eb200 8000cc4e246eb200
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add the **lag** *lag-id* parameter. |
| 08.0.70 | This command was modified to display path cost values from the IEEE 802.1D 2004 standards. |

# show span designated-protect

Displays a list of all ports that are not allowed to go into the designated forwarding state.

## Syntax

**show span designated-protect**

## Modes

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

## Examples

The following example indicates that the designated forwarding state is disallowed for interfaces 2/1/7, 2/1/19, and 2/2/3.

```
device(config)# show span designated-protect
Designated Protection Enabled on:
Ports: (U2/M1)   7 19
Ports: (U2/M2)   3
```

## History

| Release version | Command history |
|---|---|
| 07.3.00g | This command was introduced. |

# show spx

Displays information on Switch Port Extender (SPX) topology.

## Syntax

**show spx** [ **cb-port** *unit/slot/port* ] | **lag** | **mecid** | **multicast** | **pe-id** *identifier* | **pe-group** *name* | **ring** { **all** | **chain** [ *unit/slot/port* ] | *ID* } | **zero-ipc** ]

## Command Default

Without parameters, the **show spx** command displays overall SPX topology.

## Parameters

**cb-port** *unit/slot/port*
PE chain attached to this CB port, identified by port, for which information is displayed. To display information for an entire LAG, any port in the LAG can be entered.

**lag**
Displays SPX LAG information.

**mecid**
Displays 802.1br multicast E-CID information.

**multicast**
Displays Multicast information.

**pe-id** *identifier*
Specifies the PE chain containing this PE.

**pe-group** *name*
Specifies the PE group for which the PE chain is displayed.

**ring**
Specifies the PE ring or rings for which information is displayed.

**all**
Displays information for all PE rings in the Campus Fabric domain.

**chain**
Displays information on PE chains in the Campus Fabric domain.

*unit/slot/port*
Specifies the connecting port that identifies the PE chain.

*ID*
Specifies the ID of the PE ring, in decimal format, for which information is displayed.

**zero-ipc**
Displays statistics for packets used in the Zero-touch provisioning (ZTP) or SPX interactive-setup process.

# Modes

CB configuration mode

PE mode

PE configuration mode

# Usage Guidelines

The command can be issued from a CB or a PE; however, the display from a PE is limited to information about the PE itself.

If the CB fails to reload a new unit, check the Reliable-mail success count.

# Command Output

The **show spx** command displays the following information:

| Output field | Description |
|---|---|
| ID | SPX unit ID (CB or PE; PE units are numbered from 17 to 56). |
| "S" or "D" | Static or Dynamic. Indicates whether the configuration has been saved to memory. Dynamic configurations are lost when the unit is removed. |
| Type | FastIron device model (SKU). |
| Role | Lists role unit plays in SPX system, from among these possibilities: active (active controller in the CB stack), member (CB stack member), standby (standby controller for the CB stack), or spx-pe (PE unit). |
| Mac address | MAC address of the unit. |
| Pri | Priority of the unit. |
| State | Displays state of unit, from among these possibilities: Local (unit from which the command was entered), remote (unit is not local), or reserve (indicates a reserved, rather than an active, unit). |
| Comment | Ready (unit is operational).<br>Synchronizing (output example: 21 S ICX7450-24G spx-pe cc4e.248b.a448 N/A remote Synchronizing (st=13)). |

# Examples

The following example displays information for a CB.

```
device(config)# show spx
T=2h57m47.1: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7750-48XGF active  cc4e.2438.7e80 128 local   Ready
2  S ICX7750-48XGF standby cc4e.246e.cd80 128 remote  Ready
17 S ICX7450-48P   spx-pe  cc4e.248b.da60 N/A remote  Ready
18 S ICX7450-24G   spx-pe  cc4e.246c.e3f8 N/A remote  Ready
19 S ICX7450-24G   spx-pe  cc4e.246c.e420 N/A remote  Ready

    standby      active
     +---+        +---+
  2/4| 2 |2/1--2/1| 1 |2/4
     +---+        +---+

            +----+        +----+
 2/1/41--2/4| 17 |1/1==1/1| 18 |
            +----+        +----+

            +----+
  2/1/7--2/3| 19 |
            +----+
```

The following example shows output for a CB. The output includes warning messages that appear when the system detects a mismatch in port number for an operational port in any link.

```
device# show spx
T=13m35.0: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7750-20QXG standby cc4e.2438.7280 128 remote  Ready
2  S ICX7750-20QXG member  748e.f8f9.6300   0 remote  Ready
3  S ICX7750-20QXG active  cc4e.2438.7500 128 local   Ready
21 S ICX7450-48GF  spx-pe  0000.0000.0000 N/A reserve
24 S ICX7450-48G   spx-pe  cc4e.248b.77b0 N/A remote  Ready
56 S ICX7450-48GF  spx-pe  cc4e.246c.f190 N/A remote  Ready

    active       standby
     +---+        +---+        +---+
 =2/1| 3 |2/4--2/1| 1 |2/4==2/1| 2 |2/4=
  |  +---+        +---+        +---+   |
  |                                     |
  |-------------------------------------|

            +----+        +----+
  3/1/5==3/1| 24 |2/1==2/1| 56 |2/3=
            +----+        +----+

**** Warning! 1 link has non-matching port number or UP status.
Please ignore this warning if it is during PE formation or transit time.
same # 2, but diff UP #: 1 -- 2, link: 3/1/5 3/1/8 -- 24/3/1 24/4/1
```

The following example displays information for a PE when the command is entered locally on the PE unit.

```
[PE]local-id@device# show spx
T=20h30m52.8: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
24 S ICX7450-48G   spx-pe  cc4e.248b.77b0 N/A local   Ready

          +----+
       =2/1| 24 |3/1=
           +----+
```

The following example displays information for an SPX PE group.

```
device(config)# show spx pe-group GROUP1
Show PEs attached to pe-group GROUP1 (port 2/1/41)
T=2h56m13.3: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7750-48XGF active  cc4e.2438.7e80 128 local   Ready
2  S ICX7750-48XGF standby cc4e.246e.cd80 128 remote  Ready
17 S ICX7450-48P   spx-pe  cc4e.248b.da60 N/A remote  Ready
18 S ICX7450-24G   spx-pe  cc4e.246c.e3f8 N/A remote  Ready

    standby      active
     +---+        +---+
  2/4| 2 |2/1--2/1| 1 |2/4
     +---+        +---+

            +----+        +----+
 2/1/41--2/4| 17 |1/1==1/1| 18 |
            +----+        +----+
```

The following example displays information for all PE rings in a Campus Fabric domain.

```
ICX7750-48F Router# show spx ring all

  ------------------------------------------------------------------------------------
  Ring Id |FSM State |CB port  Lag  |Remote CB port  Lag  |Log Block Link
  ------------------------------------------------------------------------------------
  1        ACTIVE     1/1/2    3072  1/1/3           3073   PE29x--xCB1/1/3
  2        ACTIVE     3/1/19   3076  3/1/45          3078   CB3/1/19x--xPE26
  3        ACTIVE     2/1/12   3075  3/1/31          3077   PE27x--xPE28
  4        ACTIVE     1/1/5    3079  2/1/9           3074   PE19x--xPE20
```

The following example narrows the output to a specific ring (ring IDs can be derived from the **show spx ring all** command).

```
ICX7750-48F Router# show spx ring 1

  ------------------------------------------------------------------------------------
  Ring Id |FSM State |CB port  Lag  |Remote CB port  Lag  |Log Block Link
  ------------------------------------------------------------------------------------
  1        ACTIVE     1/1/2    3072  1/1/3           3073   PE29x--xCB1/1/3
```

The following example shows output for all PE chains in the domain.

```
ICX7750-48F Router# show spx ring chain

-------------------------------------------------------------------------------------------------------
  CB Port    Lag  | Epoch |Ring    Id      FSM State|Remote CB port  Lag  |         Chain
  ------------------------------------------------------------------------------------------------------
  1/1/2      3072  57      YES     1       ACTIVE    1/1/3           3073   PE23--PE24--PE29x--xCB1/1/3


          PE chain information connecting to CB port 1/1/2 [Lag 3072]
             ---------------------------------------------------------------------------
             PE Id | Epoch | FSM state |Uplink port|Casc port|Log Block|Prev PE|Next PE
             ---------------------------------------------------------------------------
             PE23   57      ACTIVE      23/2/2      23/2/1    NO        --      PE24
             PE24   57      ACTIVE      24/2/1      24/1/1    NO        PE23    PE29
             PE29   57      ACTIVE      29/1/1      29/2/3    YES       PE24    x--x


          PE chain information connecting to remote CB port 1/1/3 [Lag 3073]
             ---------------------------------------------------------------------------
             PE Id | Epoch | FSM state |Uplink port|Casc port|Log Block|Prev PE|Next PE
             ---------------------------------------------------------------------------
```

The following example narrows the output to a PE chain from one CB port (does not have to be the local CB).

```
ICX7750-48F Router# show spx ring chain 1/1/2    <-- This port number can be a local CB Port or a
remote CB port.

  ------------------------------------------------------------------------------------------------
  CB Port    Lag   | Epoch |Ring    Id      FSM State|Remote CB port  Lag  |         Chain
  ------------------------------------------------------------------------------------------------
  1/1/2      3072  57      YES     1       ACTIVE    1/1/3           3073   PE23--PE24--PE29x--xCB1/1/3


          PE chain information connecting to CB port 1/1/2 [Lag 3072]
             ---------------------------------------------------------------------------
             PE Id | Epoch | FSM state |Uplink port|Casc port|Log Block|Prev PE|Next PE
             ---------------------------------------------------------------------------
             PE23   57      ACTIVE      23/2/2      23/2/1    NO        --      PE24
             PE24   57      ACTIVE      24/2/1      24/1/1    NO        PE23    PE29
             PE29   57      ACTIVE      29/1/1      29/2/3    YES       PE24    x--x


          PE chain information connecting to remote CB port 1/1/3 [Lag 3073]
             ---------------------------------------------------------------------------
             PE Id | Epoch | FSM state |Uplink port|Casc port|Log Block|Prev PE|Next PE
             ---------------------------------------------------------------------------
```

The following example displays zero-IPC information. If the success count is smaller than the send count in the **target MAC** row, it means some reliable messages are lost. This could affect reloading new units as PEs.

```
ICX7750-20Q Router# show spx zero-ipc
!
!
!
Send message types:
    [5]=2,            [6]=4,            [9]=1,
Recv message types:
    [3]=30,           [6]=4,

Statistics:
    send pkt num          :         6,    send pkt-msg num      :         6,
    recv pkt num          :        34,    send msg num          :         6,
    recv msg num          :        34,    pkt buf alloc         :         6,

    Reliable-mail        send  success  receive   duplic  T (us)
    target MAC              1        1        0        0     6532 <-----------------
    unrel target MAC        1                 0

Possible errors:
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.50 | PE ring and zero-IPC parameters were added. |
| 08.0.61 | show spx csp and show spx debug options were added and documented on separate command pages. |

# show spx connections

Displays information on SPX port connections.

## Syntax

**show spx connections**

## Modes

Privileged EXEC mode (from CB)

## Usage Guidelines

Use the **show spx connections** command to determine which SPX port from one unit is connected to which SPX port on another unit.

Use the **show spx connections** command to determine whether the data flow on an SPX port is unidirectional (arrows with single head, for example, -->) or bidirectional (arrows with dual heads, for example, <--->).

For details on the connections between each device in an SPX domain, including device status and domain connection topology, use the **show spx** command.

## Examples

The following example displays **show spx** output and **show spx connections** output for the same Campus Fabric domain. Connections to two ICX 7450 units serving as active PEs (IDs 17 and 23) are detailed in the **show spx connections** output.

```
ICX7750-48F Router# show spx
T=7m34.5: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7750-48XGF active  cc4e.2438.a580 128 local   Ready
2  S ICX7750-48XGF standby cc4e.2438.8d80 128 remote  Ready
17 S ICX7450-24G   spx-pe  cc4e.246c.e2b8 N/A remote  Ready
18 S ICX7450-48G   spx-pe  0000.0000.0000 N/A reserve
23 S ICX7450-48G   spx-pe  cc4e.246c.ea50 N/A remote  Ready

    standby      active
     +---+        +---+
     | 2 |2/1--2/1| 1 |2/4
     +---+        +---+
           +-----+        +-----+
  1/1/1--2/1| 17 |2/3==2/3| 23 |2/1--1/1/6
            +-----+        +-----+


ICX7750-48F Router# show spx connections
Probing the topology. Please wait ...
ICX7750-48F Router# Discovered following spx connections...
Link 1: # of ports in lag = 1
       1: 1/1/1 <--> 17/2/1    <-- All links shown here are bi-directional.
Link 2: # of ports in lag = 2
       1: 17/2/3 <--> 23/2/3
       2: 17/2/4 <--> 23/2/4
Link 3: # of ports in lag = 1
       1: 23/2/1 <--> 1/1/6
Discovery complete
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# show spx csp

Displays Control and Status Protocol (CSP) information for the specified PE or all PE units.

## Syntax

**show spx csp** { *pe-id* | **all** | **distributed** { *pe-id* | **all-pe** | **units** *list* } [ **detail** ] }

**show spx csp events**

**show spx csp events** [ **all** | **distributed** { *pe-id* | **all-pe** | **units** *list* } ]

**show spx csp events misc** [ **distributed** ] { *pe-id* | **all-pe** | **units** *list* }

## Parameters

*pe-id*
Specifies the number of the attached PE from which CSP information is to be obtained.

**all**
Displays all CSP information, including miscellaneous events.

**distributed**
Specifies that debug information is to originate from a particular PE unit, a list of PE units, or all PE units (rather than from the data stored at the CB).

**all-pe**
Specifies that CSP information will be **obtained** from all PE units.

**units** *list*
Lists the units from which CSP information is to be obtained.

**detail**
Displays detailed information for specified items.

**events**
Displays information on CSP events.

*pe-id*
Displays CSP event information for the PE number specified.

**all**
Displays all CSP events.

**misc**
Displays miscellaneous CSP event information.

## Modes

Privileged EXEC mode

# Usage Guidelines

It is advisable to check CSP protocol issues on the CB and on remote PE units. Use the keyword **distributed** to display CSP information directly from PE units. Without the keyword, CSP protocol information from the CB is displayed.

As a complement to **show spx csp pe-log** output, refer to the troubleshooting command **show spx zero-touch log**.

# Command Output

The **show spx csp** command displays the following information:

| Output field | Description |
|---|---|
| Unit ID MAC | MAC address of unit specified in show command |
| CSP Oper: | CSP Operational (yes or no ) |
| Attach time: | Up time recorded by the CB since this PE joined the network |
| Up time: | Time unit is active (hours, minutes, seconds) |
| CB SPX LAG ID: | Information on CB SPX LAG, including:<br>• Network identifier for the LAG to which unit belongs<br>• IPC/ECP Port: SPX port on CB unit through which IPC/ECP messages are sent to the designated PE<br>• Current state: Current state of LAG (up or down) |
| PE SPX Uplink Port: | SPX port on the PE unit (identified as unit/slot/port) through which it connects to the CB, current status (Should always be up.) |
| Number of Traffic Class: | Number of QoS traffic classes supported on the designated PE |
| Priority Flow Control: | Priority flow control setting (yes or no) |
| CSP control ECID handshake complete: | Control channel communication established between CB and designated PE (yes or no) |
| CSP control ECID: | Identifier of E-channel allocated to the designated PE |
| CSP Alternate control ECID: | Alternate E-channel allocated to the designated PE, when PE is attached in a ring. Becomes the active ECID when the PE cannot reach the CB through its uplink port (typically, when the ring is broken). |
| PE is in a Ring | Appears only if PE is part of an SPX ring. Displays status of the ring (active or broken), Logical Block: (indicates whether PE is a logical end point in a ring; possible values: 1 or 0) |
| CB Alternate Spx Lag id: | For PEs in a ring, displays the alternate ID SPX LAG number and related information. |
| Total number of configured ports | Total number of ports configured on the designated PE. |
| CSP number of allocated ECIDS (VPs created, excl Control VP): | Number of ECIDs allocated to the designated PE (for data ports) |
| List of ports | Information on all SPX ports configured on the designated PE (unit/slot/port), including:<br>• ecid: ECID associated with the port.<br>• spx-port-type: Type of SPX port (on CB: Host, Uplink, or Cascade; on PE: Host, Uplink, Cascade (Init), Cascade (Forward), or Cascade (Block))<br>• cascade-port: On CB: SPX cascade port through which this data port can be reached; on PE: uplink port through which the designated PE can reach the CB<br>• cascade-lag: SPX LAG ID associated with cascade port |
| CSP last TX Trans ID, last RX Trans ID | Last sent transmission ID, last received transmission ID |
| ECP transmission statistics | ECP transmission statistics, including:<br>• txErrors: ECP transmission errors<br>• sequence:<br>• firstSeq:<br>• lastSeq: |

| Output field | Description |
|---|---|
| | • firstAckIdx:<br>• ackIdx:<br>• tPause:<br>• state:<br>• ackTimer:<br>• syncTimer:<br>• sync_cnt:<br>• need_cleanup: |
| Next PE: | Next PE in the ring or chain (if any) |
| PE Spx downlink Port: | When there is a next PE, port ID (or first reachable port ID for an SPX LAG) of the SPX downlink between the designated PE and the next PE |
| Previous PE: | ECP information for previous PE in chain (if any). |
| Local CSP Major version | For local peer (CB or PE), Ruckus CSP software version |
| Peer CSP Major version | Ruckus CSP software version for remove peer (CB or PE) |
| Oper CSP Major version | Operating CSP version (lowest shared version of peer and remote versions) |
| Msg-ID list | Message statistics by Message ID, including:<br>• Msg-Name: Name of message type associated with Msg-ID<br>• Local (ver, size): Local peer (CSP software version, message size in Bytes)<br>• Peer (ver, size): Remote peer (CSP software version, message size in Bytes)<br>• Oper (ver, size): Operating CSP software (version, message size in Bytes)<br>• Up_Conv: (Yes or no; indicates whether conversion is needed (from higher to lower or operating version) when CSP messages are received from the remote peer<br>• Down_Conv: (Yes or no; indicates whether conversion is needed (from higher to lower or operating version) when CSP messages are sent to the remote peer |

# Examples

The following example provides detailed CSP information for PE unit 17.

```
ICX7750-48F Router# show spx csp 17 detail

PE 17 MAC: cc4e.246c.e2b8
   CSP Oper: yes, Attach time: 51.5, up time: 3 hour(s) 14 minute(s) 53 second(s)
   CB Spx Lag id: 3073, cur state up, IPC/ECP Port: 1/1/1
   PE Spx Uplink Port: 17/2/1, cur state up
   Number of Traffic Class: 8
   Priority Flow Control: no
   CSP control ECID handshake complete: yes
   CSP control ECID: 75
   CSP Alternate control ECID: 76
   PE is in Ring (Status: Active), Logical Block: 0 <-- Information appears for PE in ring topology
only.
   CB Alternate Spx Lag id: 3072, cur state up, IPC/ECP Port: 1/1/6
   Total number of configured ports: 30
   CSP number of allocated ECIDs (VPs created, excl Control VP): 29
     port        ecid    spx-port-type   cascade-port    cascade-lag
     ------------------------------------------------------------
     17/1/1         1             Host         1/1/1        3073
     17/1/2         2             Host         1/1/1        3073
     17/1/3         3             Host         1/1/1        3073
     17/1/4         4             Host         1/1/1        3073
     17/1/5         5             Host         1/1/1        3073
     17/1/6         6             Host         1/1/1        3073
     17/1/7         7             Host         1/1/1        3073
     17/1/8         8             Host         1/1/1        3073
     17/1/9         9             Host         1/1/1        3073
     17/1/10       10             Host         1/1/1        3073
     17/1/11       11             Host         1/1/1        3073
     17/1/12       12             Host         1/1/1        3073
     17/1/13       13             Host         1/1/1        3073
     17/1/14       14             Host         1/1/1        3073
     17/1/15       15             Host         1/1/1        3073
     17/1/16       16             Host         1/1/1        3073
     17/1/17       17             Host         1/1/1        3073
     17/1/18       18             Host         1/1/1        3073
     17/1/19       19             Host         1/1/1        3073
     17/1/20       20             Host         1/1/1        3073
     17/1/21       21             Host         1/1/1        3073
     17/1/22       22             Host         1/1/1        3073
     17/1/23       23             Host         1/1/1        3073
     17/1/24       24             Host         1/1/1        3073
     17/2/1        75           Uplink         1/1/1        3073
     17/2/2        75           Uplink         1/1/1        3073
     17/2/3        76          Cascade         1/1/6        3072
     17/2/4        53             Host         1/1/1        3073
     17/3/1        60             Host         1/1/1        3073
     17/4/1        70             Host         1/1/1        3073
   CSP last Tx Trans ID=7, last Rx Trans ID=3
   ECP txErrors=0, sequence=11 firstSeq=11 lastSeq=10 firstAckIdx=0 ackIdx=0, tPause 0, state 0
   ECP ackTimer=0, syncTimer=0 sync_cnt=0 need_cleanup=0
   Next PE: 23
   PE Spx downlink Port: 17/2/3, cur state up
   Previous PE: None
   Local CSP Major version is 1 Minor version 1
   Peer  CSP Major version is 1 Minor version 1
   Oper  CSP Major version is 1 Minor version 1
    Msg-Id  Msg-Name                  Local(ver,size) Peer(ver,size) Oper(ver,size) Up_Conv Down_Conv
    ------------------------------------------------------------------------------------------------
    00      unknown (00) tlv          01,007          01,007         01,007         no       no
    01      cmd tlv                   01,009          01,009         01,009         no       no
    02      resource cap tlv          01,042          01,042         01,042         no       no
    03      port param tlv            01,160          01,160         01,160         no       no
    04      port array tlv            01,100          01,100         01,100         no       no
    05      vid array tlv             01,001          01,001         01,001         no       no
    06      port status tlv           01,016          01,016         01,016         no       no
    07      stats tlv                 01,000          01,000         01,000         no       no
```

The following example shows Control and Status Protocol (CSP) information received directly from PE units 23 and 29 as indicated by the use of the keyword **distributed** on the command line. Without the keyword, CSP information from the CB is displayed.

```
ICX7750-48F Router# show spx csp distributed units 23 29

*********************
Response from PE 23:
*********************

   CSP Oper: yes, Attach time: 1m22.0, up time: 11 hour(s) 48 minute(s) 38 second(s)
   PE Spx Lag id: 2, cur state up, IPC/ECP Port: 23/2/3
   Number of Traffic Class: 8
   Priority Flow Control: no
   CSP control ECID handshake complete: yes
   CSP control ECID: 555
   CSP Alternate control ECID: 556
   PE is in Ring (Status: Active), Logical Block: 1
   Total number of configured ports: 566
   CSP number of create port requests sent: 54
   CSP last Tx Trans ID=3, last Rx Trans ID=6
   ECP txErrors=0, sequence=10 firstSeq=10 lastSeq=9 firstAckIdx=0 ackIdx=0
   Next PE: None
   Previous PE: None
   Local CSP Major version is 1 Minor version 1
   Peer  CSP Major version is 1 Minor version 1
   Oper  CSP Major version is 1 Minor version 1

*********************
Response from PE 29:
*********************

   CSP Oper: yes, Attach time: 1m13.6, up time: 11 hour(s) 48 minute(s) 44 second(s)
   PE Spx Lag id: 2, cur state up, IPC/ECP Port: 29/2/3
   Number of Traffic Class: 8
   Priority Flow Control: no
   CSP control ECID handshake complete: yes
   CSP control ECID: 1035
   CSP Alternate control ECID: 1036
   PE is in Ring (Status: Active), Logical Block: 0
   Total number of configured ports: 566
   CSP number of create port requests sent: 54
   CSP last Tx Trans ID=3, last Rx Trans ID=9
   ECP txErrors=0, sequence=13 firstSeq=13 lastSeq=12 firstAckIdx=0 ackIdx=0
   Next PE: None
   Previous PE: None
   Local CSP Major version is 1 Minor version 1
   Peer  CSP Major version is 1 Minor version 1
   Oper  CSP Major version is 1 Minor version 1
```

The following example shows output on CSP events directly from PE unit 17 as indicated by use of the keyword **distributed** in the command line. In the example, PE port 17/1/1 has received a **loopback enable** command from the CB, and the PE has later disabled the loopback.

```
ICX7750-48F Router# show spx csp events distributed 17
*********************
Response from PE 17:
*********************
PE 0: port 17/1/1 loopback disable pass (13 minute(s) 58 second(s) )
[stack: 002dffdc 0030dd90 002f36b4 002eaee8 002ebc2c 002ea064 00324a3c 002c53dc 009f9740 00ee16c0
00ee24b4 00c0acc8]
…
<snip>
…
PE 0: port 17/1/1 loopback enable pass (13 minute(s) 58 second(s) )
[stack: 002dfdd0 0030dd90 002f36b4 002eaee8 002ebc2c 002ea064 00324a3c 002c53dc 009f9740 00ee16c0
00ee24b4 00c0acc8]
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.61 | The distributed parameter was added. |

# show spx debug

Displays debug information for an SPX port, an SPX LAG, or a set of ports or LAGs.

## Syntax

**show spx debug port** { *unit/slot/port* } [ **distributed** ]

**show spx debug port all** [ **distributed** { *pe-id* | **all-pe** | **units** *list* } ] ]

**show spx debug lag** { *lag-id* | **all** } [ **distributed** { *pe-id* | **all-pe** | **units** *list* } ]

## Parameters

**lag**

Specifies output as SPX LAG information.

*lag-id*

Identifies the SPX LAG

**all**

Includes IDs and information for all SPX LAGs in the output.

**port**

Displays information for the specified port.

*unit/slot/port*

Identifies the SPX port for which information is displayed.

**all**

Displays information for all SPX ports.

**distributed**

Specifies that debug information is to originate from a particular PE unit, a list of PE units, or all PE units (rather than from the data stored at the CB).

*pe-id*

Specifies the attached PE from which information is to be obtained.

**all-pe**

Specifies that information will be obtained from all PE units.

**units** *list*

Lists the units from which debug information is to be obtained.

## Modes

Privileged EXEC mode

## Command Output

The **show spx debug** command displays the following information:

| Output field | Description |
|---|---|
| spx-lag ID | Number of the SPX LAG. The SPX ports in the LAG are listed. |
| Port | SPX port for which information is displayed. |
| PortExtDb Index | Database active on the PE unit associated with the port. |
| Port type | Type of port by number:<br>1. Cascade<br>2. Uplink<br>3. Host<br>4. Cascade Init<br>5. Cascade FWD<br>6. Cascade BLK |
| lag_id | Identifies the SPX LAG associated with this port in hardware (used to cross check LAG ID stored in software) |

# Examples

The following example provides the SPX LAG ID (columns 1 and 4), lists the ports in each SPX LAG (column 1), provides a database index number (column 2), and displays a port type for each port in the LAG (column 3). In the example, the database index for ports 1/1/1 and 2/1/1 do not match although they should.

```
ICX7750-48F Router# show spx debug lag all

spx-lag ID 3072

Port          PortExtDb Index          Port type          lag_id
1/1/1            0                       1                 3072   |<-- Database mismatch
2/1/1            f                       1                 3072   |    on 1/1/1 and 2/1/1

spx-lag ID 3073

Port          PortExtDb Index          Port type          lag_id
1/1/6            6                       1                 3073

spx-lag ID 3074

Port          PortExtDb Index          Port type          lag_id
1/1/2            ff                      1                   3074
1/1/3            ff                      1                   3074
```

The following example obtains debug information on all SPX LAGs directly from all PE units in the system. The information comes from the PE units rather than the CB when the keyword distributed is used. The command lists the LAG IDs, ECID values, and port type for all ports in each LAG.

```
ICX7750-48F Router# show spx debug lag all distributed all-pe

**********************
Response from PE 19:
**********************

spx-lag ID 1
Port          ECID          Port type          lag_id
19/1/1         236           5                   1
19/1/2         236           5                   1

spx-lag ID 2
Port          ECID          Port type          lag_id
19/2/1         235           2                   2

**********************
Response from PE 20:
**********************

spx-lag ID 1
Port          ECID          Port type          lag_id
20/1/1         315           2                   1
20/1/2         315           2                   1

spx-lag ID 2
Port          ECID          Port type          lag_id
20/1/3         316           6                   2

**********************
Response from PE 23:
**********************

spx-lag ID 1
Port          ECID          Port type          lag_id
23/1/47          556           6                   1

spx-lag ID 2
Port          ECID          Port type          lag_id
23/2/3         555           2                   2
23/2/4         555           2                   2

**********************
Response from PE 29:
**********************

spx-lag ID 1
Port          ECID          Port type          lag_id
29/2/1        1036           5                   1

spx-lag ID 2
Port          ECID          Port type          lag_id
29/2/3        1035           2                   2
29/2/4        1035           2                   2
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.61 | The command was modified to include the distributed parameter. |

# show spx mecid

Displays IEEE 802.1 BR multicast E-channel identifier (E-CID) information.

## Syntax

**show spx mecid** [ **topology** | **all** | **decimal** | **reserved** |**summary** ]

## Parameters

**topology**
Displays 802.1 BR ME-CID Topology.

**all**
Displays all non-reserved/dynamic ME-CIDs in use.

**decimal**
The ME-CID number. The range is 4096 through 16383.

**reserved**
Displays all reserved ME-CIDs in use.

**summary**
ME-CID software settings and memory usage information.

## Modes

User EXEC mode

Privileged EXEC mode

## Command Output

The **show spx mecid topology** command displays the following information:

| Output field | Description |
|---|---|
| Total Cascade Port (CP) | Displays the details of the total number of Cascade Ports. |

The **show spx mecid all** command displays the following information:

| Output field | Description |
|---|---|
| MECID | The ME-CID number. |
| PEs | The number of port extender units. |
| VPs | The number of virtual ports. |
| AW | The number of |
| State | The FSM state - either created TX Pending, deleted TX Pending, deleted (acknowledgment waiting) or created (stable state). |
| Shr | Whether the ME-CID is shared or not. |
| Elements | The element numbers. |

The **show spx mecid decimal** command displays the following information:

| Output field | Description |
|---|---|
| MECID state | The state of the ME-CID - either created TX Pending, deleted TX Pending, deleted (acknowledgment waiting) or created (stable state). |
| VP state | The state of the virtual port. |
| MECID | Displays details of the specified ME-CID. |

The **show spx mecid summary** command displays the following information:

| Output field | Description |
|---|---|
| alloc | Number of nodes of data that are currently allocated in memory. |
| in-use | Number of allocated nodes in use. |
| avail | Number of allocated nodes not in use. |
| get fail | Number of allocation failures for this node. |
| limit | Maximum number of nodes that can be allocated for a data structure. This may or may not be configurable, depending on the data structure. |
| get-mem | Number of successful allocations for this node. |
| size | The size of the node in bytes. |
| init | Number of nodes that are allocated during the time of initialization. |

# Examples

The following example displays the ME-CID topology.

```
device# show spx mecid topology
Total Cascade Port(CP): 3
   1. CP-TR(e2/3/5)    :-->[e56/3/1]PE_56

   2. CP-TR(e1/1/47)   :-->[TR(e17/2/1)]PE_17[TR(e17/1/21)]-->
                          [TR(e18/1/1)]PE_18[e18/1/2]-->[e19/1/22]PE_19[TR(e19/2/1)]--
>[TR(e20/1/21)]PE_20

   3. CP-TR(e2/3/1)    :-->[TR(e21/3/1)]PE_21
```

The following example displays information about all the ME-CIDs.

```
device# show spx mecid all
FSM-State : CREATE_P - created TX Pending, DELETE_P - deleted TX Pending
            DELETE_AW - Ack Waiting, CREATED - Stable State
Total MECID Allocated 360
-----------------------------------------------------
SNo   MECID PEs VPs  AW  State      Shr   Elements
-----------------------------------------------------
1     4096  10  416  0   CREATED    0     {16/4/64 17/1/1 17/1/2 17/1/3 17/1/4 1
7/1/5 17/1/6 ...}
2     4097  0   0    0   CREATED    0     {}
3     4098  0   0    0   CREATED    0     {}
4     4099  0   0    0   CREATED    0     {}
5     4100  0   0    0   CREATED    0     {}
6     4101  0   0    0   CREATED    0     {}
7     4102  0   0    0   CREATED    0     {}
8     4103  0   0    0   CREATED    0     {}
9     4104  0   0    0   CREATED    0     {}
10    4105  0   0    0   CREATED    0     {}
11    4106  0   0    0   CREATED    0     {}
12    4107  0   0    0   CREATED    0     {}
13    4108  0   0    0   CREATED    0     {}
14    4109  0   0    0   CREATED    0     {}
15    4110  0   0    0   CREATED    0     {}
16    4111  0   0    0   CREATED    0     {}
17    4112  0   0    0   CREATED    0     {}
```

The following example displays the output of the **show spx mecid reserved** command.

```
device# show spx mecid reserved
FSM-State : CREATE_P - created TX Pending, DELETE_P - deleted TX Pending
            DELETE_AW - Ack Waiting, CREATED - Stable State
Total MECID Allocated 360
-----------------------------------------------------
SNo   MECID PEs VPs  AW  State      Shr   Elements
-----------------------------------------------------
1     4096  10  416  0   CREATED    0     {16/4/64 17/1/1 17/1/2 17/1/3 17/1/4 1
7/1/5 17/1/6 ...}
2     4097  0   0    0   CREATED    0     {}
3     4098  0   0    0   CREATED    0     {}
4     4099  0   0    0   CREATED    0     {}
5     4100  0   0    0   CREATED    0     {}
6     4101  0   0    0   CREATED    0     {}
7     4102  0   0    0   CREATED    0     {}
8     4103  0   0    0   CREATED    0     {}
9     4104  0   0    0   CREATED    0     {}
10    4105  0   0    0   CREATED    0     {}
11    4106  0   0    0   CREATED    0     {}
12    4107  0   0    0   CREATED    0     {}
13    4108  0   0    0   CREATED    0     {}
14    4109  0   0    0   CREATED    0     {}
15    4110  0   0    0   CREATED    0     {}
16    4111  0   0    0   CREATED    0     {}
17    4112  0   0    0   CREATED    0     {}
18    4113  0   0    0   CREATED    0     {}
19    4114  0   0    0   CREATED    0     {}
```

The following example displays information about ME-CID 16383.

```
device# show spx mecid 16383
MECID State : CREATE_P - created Tx Pending, DELETE_P - deleted Tx Pending
              DELETE_AW - Ack Waiting, CREATED - Stable State
VP State    : ADD_P - Added Tx Pending, REMOVE_P - Removed Tx waiting
            : ADDED - Stable State
MECID: 16383 Total PEs: 4, Epoch: 0, FSM State: CREATED, SetId: 0x0, AW: 0
   1. PE: 17, Total VPs: 2, Ack_waiting: 0, CP Added: Yes
      (17/1/21(cp) ADDED),  (17/1/29(cp) ADDED),
   2. PE: 18, Total VPs: 1, Ack_waiting: 0, CP Added: Yes
      (18/1/2 (cp) ADDED),
   3. PE: 19, Total VPs: 2, Ack_waiting: 0, CP Added: Yes
      (19/2/1 (cp) ADDED),  (19/2/4 (cp) ADDED),
   4. PE: 20, Total VPs: 1, Ack_waiting: 0, CP Added: No
      (20/2/4  ADDED),
```

The following example displays the ME-CID summary.

```
device# show spx mecid summary
  Manager Init          : Yes        Replication Group Sharing : No
  Reconciliation Pass   : 0          Replication Id            : 0
  ECID Partition        : Enabled    Global Timer running      : No
  ECID sharing          : Yes
  ------------------------------------------------------------------------------
                     alloc in-use  avail get-fail    limit  get-mem  size init
  ------------------------------------------------------------------------------
  MECID info          1000    360    640        0   232000     1473    66 1000
  PE info              400     60    340        0    92800      414    52  400
  VP info            10000    570   9430        0  2320000     2564    25 1000
  TX Q                1000      0   1000        0   232000     1621    12 1000
  PE Msg              1024      0   1024        0   237568     1367    56 1024

  Total memory in used: 406144 bytes
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.50 | The outputs of the **show spx mecid all** and **show spx mecid reserved** commands were modified. |

# show spx multicast cache

Displays multicast E-CID forwarding entries for the PE.

## Syntax

**show spx multicast cache** [ **ecid** ]

## Parameters

**ecid**

The multicast E-CID. The allowed range is 4096 through 16384.

## Modes

User EXEC mode

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show spx multicast cache** command displays the following information:

| Output field | Description |
|---|---|
| E-CID | Point-to-multipoint E-CID. |
| UpTime | The time elapsed since the ME-CID forwarding cache was setup on the PE unit. |
| LastUpdate | The time elapsed since this ME-CID forwarding cache was updated. |
| HWSltMsk | The bitmask used to represent the Packet Processor (PP) hardware chips on which this forward entry is programmed successfully. |
| L2MC | Hardware replication resource used to replicate traffic for the ME-CID forwarding cache to local ports. |
| SetID | Identifies the internal software resource used in sharing (optimizing). |
| Ports | The list of outgoing ports for the specific ME-CID forwarding cache. |

# Examples

The following command displays multicast E-CID forwarding entries for the PE.

```
device# show spx multicast cache
1     E-CID: 5120  UpTime: 00:03:05  LastUpdate: 00:02:29
      HWSltMsk: 0x1, L2MC: 1025(Shr), SetID: 0x31255948
      Ports 5:
        17/1/1 (00:03:05)  17/1/3 (00:02:37)  17/1/5 (00:03:05)  17/1/7 (00:03:05)
        17/1/10(00:02:29)

2     E-CID: 6451  UpTime: 00:01:31  LastUpdate: 00:00:44
      HWSltMsk: 0x1, L2MC: 1025(Shr), SetID: 0x31255948
      Ports 5:
        17/1/1 (00:01:20)  17/1/3 (00:00:44)  17/1/5 (00:01:20)  17/1/7 (00:01:31)
        17/1/10(00:01:31)

3     E-CID: 9276  UpTime: 00:00:07  LastUpdate: 00:00:07
      HWSltMsk: 0x1, L2MC: 1027(Shr), SetID: 0x31739250
      Ports 2:
        17/1/2 (00:00:07)  17/1/14(00:00:07)
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show spx multicast optimization

Displays the multicast replication resource optimization details for the PE unit.

## Syntax

**show spx multicast optimization** [ **repl-id** ]

## Parameters

**repl-id**

The multicast replication resource identifier. The allowed range is 1 through 8192.

## Modes

User EXEC mode

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show spx multicast optimization** command displays the following information:

| Output field | Description |
|---|---|
| L2MC | Hardware replication used to replicate traffic for the ME-CID forwarding cache. |
| SetId | Internal software resource used in sharing the L2MCs by the ME-CID forwarding cache entries. |
| Users | The number of ME-CID forwarding cache entries sharing the specific L2MC resource. |
| Set | The list of ports. |
| Sharability coefficient | The quantitative representation of L2MC sharing by the forwarding entries. For example, if the system has four forwarding entries, and they are using a single L2MC, then the sharability coefficient is 100%. But if they use 2 L2MCs together, then the sharability coefficient is 50%. |

## Examples

The following example displays the multicast replication resource optimization details for the PE unit.

```
device# show spx multicast optimization 2
Total L2MCs Allocated:    2; Available:  7166; Failed:   0
Index    L2MC          SetId         Users  Set
   1.    1027          0x31739250        1 {<17/1/14>,<17/1/2>,}
   2.    1025          0x31255948        2 {<17/1/10>,<17/1/7>,<17/1/5>,<17/1/3>,
                                            <17/1/1>,}
Sharability Coefficient:  50%
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show spx multicast resource

Displays multicast memory pool details for the PE unit.

## Syntax

**show spx multicast resource**

## Modes

Privileged EXEC mode

## Usage Guidelines

## Command Output

The **show spx multicast resource** command displays the following information:

| Output field | Description |
|---|---|
| alloc | Number of nodes of data that are currently allocated in memory. |
| in-use | Number of allocated nodes in use. |
| avail | Number of allocated nodes not in use. |
| get-fail | Number of allocation failures for this node. |
| limit | Maximum number of nodes that can be allocated for a data structure. This may or may not be configurable, depending on the data structure. |
| get-mem | Number of successful allocations for this node. |
| size | The size of the node in bytes. |
| init | Number of nodes that are allocated during the time of initialization. |

## Examples

The following command displays the multicast memory pool details for the PE unit.

```
device# show spx multicast resource
                    alloc in-use  avail get-fail    limit    get-mem  size  init
fwd mcache             64      1     63        0    14848        1     68    64
tx port info          256     54    202        0    59392       54     23    256

Total memory used: 10240 bytes
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show spx pe-port-vlan-resources

Displays VLAN resources per PE port for an SPX (Campus Fabric) system.

## Syntax

**show spx pe-port-vlan-resources**

## Modes

All modes

## Command Output

The **show spx pe-port-vlan-resources** command displays the following information:

| Output field | Description |
|---|---|
| Reserved number of VLANs for each PE port | 4 (fixed value). Number of VLANs for which each PE port is guaranteed membership. |
| Maximum number of VLANs a PE port can be added to | Maximum allowable VLANs per PE port as configured with the max-vlans-per-pe-port (Valid range is 5 through 1024). |
| Configured PE VLAN entries | Number of VLANs configured for PEs in the SPX system. |
| PE VLAN Global Pool size | 4096. Global pool of available PE port-to-VLAN assignments, not including four VLAN assignments already reserved for each PE port. |
| Entries used in PE VLAN Global Pool | Number of configured entries in the global PE VLAN pool. The number may be oversubscribed. |
| Entries available in PE VLAN Global Pool | From 4096, number of remaining entries in the PE VLAN pool. |
| Ports with more than reserved number of VLANs | Number of PE ports that have more than 4 VLANs configured. |
| Configuration failures due to hash collisions | Number of PE port VLAN configuration rejections due to table space collision at or near maximum scaling. |

## Examples

The following example shows output for the **show spx pe-port-vlan-resources** command.

```
ICX7750-48F Router(config-spx-cb)# show spx pe-port-vlan-resources
PE VLAN Global Pool Resource Usage:
============================================================
Reserved number of VLANs for each PE port : 4
Maximum number of VLANs a PE port can be added to : 1023
Configured PE VLAN entries : 4483
PE VLAN Global Pool size : 4096
Entries used in PE VLAN Global Pool : 4096
Entries available in PE VLAN Global Pool : 0
Ports with more than reserved number of VLANs : 33
Configuration failures due to hash collisions : 0
============================================================
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# show spx zero-touch ipc

Displays statistics for Inter-Processor Communication (IPC) used by zero-touch-enable and spx interactive-setup processes.

## Syntax

**show spx zero-touch ipc**

## Modes

Privileged EXEC mode

## Command Output

The **show spx zero-touch ipc** command displays the following information:

| Output field | Description |
|---|---|
| V1 | Version 1 |
| src | Source MAC address of egress IPC packets (MAC address of this unit). |
| max_pkt_size | Maximum packet size allowed in network. |
| recv | Total packets received. |
| send | Total packets sent. |
| Send message types | Totals for each type of message sent, displayed in the following format: [ x ] = total, where "x" represents the number of a message type from the **Message types have callbacks** list. |
| Recv message types | Totals for each type of message received, displayed in the format [ x ] = total, where "x" represents the number of a message type from the **Message types have callbacks** list. |
| Statistics: | |
| send pkt num | Total number of packets sent. One packet may contain multiple messages. |
| recv pkt num | Total number of packets received. One packet may contain multiple messages. |
| recv msg num | Total number of received messages related to Zero-touch or SPX interactive-setup (see list of types in output). |
| send pkt-msg num | Total number of packets containing Zero-touch or SPX interactive-setup messages sent. |
| send msg num | Total number of Zero-touch or SPX interactive-setup messages sent. |
| pkt buf alloc | Packet buffer allocation size. |
| Reliable mail | Reliable-mail messages are used for essential communications, for example, to assign PE IDs or reload the system.<br>Reliable-mail message statistics for specified target types:<br>• send: number of reliable-mail messages sent<br>• success: number of successful reliable-mail messages (sent and acknowledged)<br>• receive: number of packets received<br>• duplic: number of duplicate packets sent<br>• T (US): Average time (in microseconds) between packet transmission and receipt of acknowledgment |
| target MAC | Number of reliable-mail messages sent using the MAC address as the target address. (Reliable-mail messages are retransmitted until acknowledgment is received.) |

| Output field | Description |
|---|---|
| unrel target MAC | Number of unreliable-mail messages using the MAC address as the target address. (Unreliable-mail messages are sent only once.) |
| Possible errors: | Warnings or errors detected, if any. |

# Examples

The following example indicates that this CB unit has sent 120 type 4 (zero-touch request) messages. The unit has also sent 10 type 6 (reliable-mail) messages. Reliable-mail messages are used for essential communications, for example, to assign PE IDs or reload the system.

```
ICX7750-20Q Router# show spx zero-touch ipc
V1, , src=cc4e.2438.7500, max_pkt_size=1468, recv 85, send 130
Message types have callbacks:
3: zero-touch probe                  4: zero-touch request
5: unreliable-mail                   6: reliable-mail
7: test ipc packets                  8: cmd_to_new_unit
9: KA_new_unit

Send message types:
    [4]=120,          [6]=10,
Recv message types:
    [3]=67,           [6]=15,

Statistics:
    send pkt num          :       130,    send pkt-msg num      :       130,
    recv pkt num          :        85,    send msg num          :       130,
    recv msg num          :        85,    pkt buf alloc         :       130,

    Reliable-mail        send  success  receive   duplic  T (us)
    target MAC              4        4        0        0    19555
    unrel target MAC        0                 0

Possible errors:
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. The command replaces the show spx zero-ipc command. |

# show spx zero-touch log

Displays the contents of the internal Campus Fabric Zero-touch provisioning log.

## Syntax

**show spx zero-touch log**

## Modes

Privileged EXEC mode

## Usage Guidelines

Scan or search the log for Error or Warning items for details on potential problems.

## Examples

The following example displays detailed information on **zero-touch-enable** and **spx interactive-setup** processes.

```
ICX7750-20Q Router# show spx zero-touch log
42.4516 ZTP chg_cb(old=0, new=4): I new-A, ZTP not enabled, , 1U 0P A4S0 I4A 1%
8m42.4057 init_zero_touch() init_T=5217 , 3U 0P A4S2 I4A 81%
9m46.4440 Send ZTP probes: u1, ports: 4/1/6 to 4/1/8 PEs: , 3U 0P A4S2 I4A
10m7.5115 cb_r_probe. rec#=2, load=173, inv [0] cc4e.248b.77b0,  rec#=2, exist mac=cc4e.246c.f190 <=
cc4e.248b.77b0,
  cannot overwrite, 3U 0P A4S2 I4A
21m38.4824 ZTP, 12 .5min T, cb_state = 0, diff = 201 s, diff=201 > 120 sec, trigger probe, , 3U 1P A4S2
I4A 20%
21m38.6988 Send ZTP probes: u1, ports: 4/1/6 to 4/1/8 PEs: pe19, , 3U 1P A4S2 I4A
21m46.3364 ZTP 100ms: tk=51, st=1, cb_tk=19 max_tk=20, probe, wait=2, , 3U 1P A4S2 I4A
21m46.7642 ZTP 100ms: tk=52, st=1, cb_tk=20 max_tk=20, ->GET_RES, wait=20,, 3U 1P A4S2 I4A
21m59.7897 ZTP 100ms: tk=62, st=2, cb_tk=9 max_tk=10, probe, wait=5, , 3U 1P A4S2 I4A 16%
21m59.8047 cb_r_probe. rec#=1, load=114, inv [0] cc4e.248b.77b0,  use old 0,  , 3U 1P A4S2 I4A
22m0.8002 Send cleanup 20 sec, rel=1, to (chain# res): C0 1, , 3U 1P A4S2 I4A 32%
22m0.8003 chain 0:  #1 CC4E.248B.77B0 ID=20, D0: 2/2 to 2/3, D1: 3/1 4/1
, 3U 1P A4S2 I4A
22m0.8003 Zero-touch discovers 1 chain(s). # of valid chains: 1 total=1, 3U 1P A4S2 I4A
22m0.8003 ZTP 100ms: tk=63, st=2, cb_tk=10 max_tk=10, found 1 chains, unstable#=0, , 3U 1P A4S2 I4A
22m20.7974 Add spx-ports: , 3U 1P A4S2 I4A 1%
22m20.7974 ztp_sync_cb_lag, from_T=0, S=u2 st=6, buf=NULL, do nothing , 3U 1P A4S2 I4A
22m20.7974 reset: free: X C0, X chain_cb_pe_p, , 3U 1P A4S2 I4A
24m36.0966 ZTP-speedup, spx-port 20/4/1 UP, tick=10 , 3U 2P A4S2 I4A
24m42.2637 topo chg: during ztp reload, abort.
, 3U 2P A4S2 I4A 2%
25m15.8573 ZTP, 12 .5min T, cb_state = 0, diff = 33 s, diff=33 < 120, abort, 3U 2P A4S2 I4A 1%
```

The following example indicates that port 1/1/47 links to an invalid chain that already contains a maximum number of PE units. The output also indicates a ZTP reliable mail message has not been delivered.

```
ICX7750-20Q Router# show spx zero-touch log
42.4516 ZTP chg_cb(old=0, new=4): I new-A, ZTP not enabled, , 1U 0P A4S0 I4A 1%
8m42.4057 init_zero_touch() init_T=5217 , 3U 0P A4S2 I4A 81%
9m46.4440 Send ZTP probes: u1, ports: 4/1/6 to 4/1/8 PEs: , 3U 0P A4S2 I4A
10m7.5115 cb_r_probe. rec#=2, load=173, inv [0] cc4e.248b.77b0,  rec#=2, exist mac=cc4e.246c.f190 <=
cc4e.248b.77b0, cannot overwrite, 3U 0P A4S2 I4A
21m38.4824 ZTP, 12 .5min T, cb_state = 0, diff = 201 s, diff=201 > 120 sec, trigger probe, 3U 1P A4S2
I4A 20%
21m38.6988 Send ZTP probes: u1, ports: 4/1/6 to 4/1/8 PEs: pe19, , 3U 1P A4S2 I4A
28m40.3054 *** Error! 1/1/47 links to an invalid chain: (chain length=1 + PE 21 len 5 + PE 31 len 1) =
7 > max 6
, 3U 0P A4S2 I4A 90%
28m42.4057 *** Warning! ZTP rel_mail fail:  chain 0 type=8, len=102, CPU=90%, 3U 0P A4S2 I4A 90%
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show spx zero-touch status

Indicates whether Campus Fabric Zero-touch provisioning is enabled and active.

## Syntax

**show spx zero-touch status**

## Modes

Privileged EXEC mode

## Examples

The following example shows command output for a CB unit. The output indicates that both Campus Fabric (SPX) and Zero-touch provisioning have been enabled. Two valid chains have been discovered, and three attached units have been added as PEs.

```
ICX7750-20Q Router# show spx zero-touch status
zero-touch-enable and spx cb-enable are configured. Have done 2 probes
ZTP has discovered 2 valid chains and converted 3 PEs.
zero-touch-enable period: 6 minutes. Will trigger in 1 min 53 sec
ZTP postponed due to high CPU: 0, due to topology changes: 3
```

The following example shows output from a standalone ICX 7450 with startup configuration flash memory. As indicated in command output, the unit can be converted to a PE using option 3 of the **spx interactive-setup** command.

```
ICX7450-48F Router# show spx zero-touch status
I cannot be discovered by zero-touch or spx interactive-setup option 2. reason: once had startup-
configuration flash
I can be discoverd by spx interactive-setup option 3.
zero-touch-enable is not configured.
```

The following example shows command output for a new unit. The output indicates that SPX Zero-touch provisioning has not been enabled.

```
ICX7750-20Q Router# show spx zero-touch status
zero-touch-enable is not configured.
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# show spx-mon

Gives a snapshot of the SPX system.

## Syntax

**show spx-mon** [ **history** [ **distributed** { *pe-id* | **units** *pe-list* | **all_pe** } ] | **pe-join** *unit/slot/port* ] [ **begin** | **exclude** | **include** *match-string* ]

## Parameters

**history**

Generates a history of SPX activity.

**distributed**

Obtains ECP events from the specified PE or all PEs, rather than from the history stored in the CB database.

*pe-id*

Specifies the PE by number from which the history is to be obtained.

**units** *pe-list*

Indicates that the history is to be obtained from the specified list of PE units.

**all_pe**

Indicates that the history is to be obtained from all connected PE units.

**pe-join** *unit/slot/port*

Provides PE status for the specified port.

**begin** *match-string*

Specifies where output starts, based on the pattern provided in the match string.

**exclude** *match-string*

Specifies what information to exclude from output based on the contents of the match string.

**include** *match-string*

Specifies what information to include in output based on the contents of the match string.

## Modes

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

Use the **show spx-mon** command to determine the health of the Campus Fabric domain. The command tells you if the domain is in good health, and if not, what corrective actions to take.

# Examples

The following example shows a Campus Fabric domain with two PE units in Ready state, and one PE has reserved configuration only. CLI guidance is given to troubleshoot a potential pe-join issue with the third PE unit.

```
ICX7750-48F Router# show spx-mon
spx-mon is enabled

Total 2 PE(s) attached, 2 attached PE(s) are in Ready state
Number of PEs in Configuration only:  1                      <--- Reserved configuration for 1 PE
Number of PEs in Config-mismatch   :  0
Number of PEs in Image-Mismatch    :  0

CLI: sh spx-mon pe-join <cb-cport> can be used to diagnose pe-join issues   <----- CLI help for
additional diagnosis

Active CPU Utilization
1   sec avg 1 percent busy
5   sec avg 1 percent busy
60  sec avg 1 percent busy
300 sec avg 1 percent busy

PE CPU Utilization:           Normal
Spx Interface Utilization:    Very low
Spx Interface Errors:         None
PE User Port Errors:          None
```

The following example indicates a configuration mismatch, as port 2/1/1 has no matching PE database. The CLI guidance given is to use **show spx debug lag all** to obtain more information.

```
ICX7750-48F Router# show spx-mon
spx-mon is enabled

Total 2 PE(s) attached, 2 attached PE(s) are in Ready state
Number of PEs in Configuration only:  1
Number of PEs in Config-mismatch   :  0
Number of PEs in Image-Mismatch    :  0

CLI: sh spx-mon pe-join <cb-cport> can be used to diagnose pe-join issues

Active CPU Utilization
1   sec avg 1 percent busy
5   sec avg 1 percent busy
60  sec avg 7 percent busy
300 sec avg 3 percent busy

Port 2/1/1 doesn't have matching PE db                       <--- Error condition indicated for Port
2/1/1.
 CLI sh spx debug lag all can be used for more information   <--- CLI help for additional diagnosis.
PE CPU Utilization:           Normal
Spx Interface Utilization:    Very low
Spx Interface Errors:         None
PE User Port Errors:          None
ICX7750-48F Router#
!!! Temperature is over warning level on stack unit 23 !!!

SYSLOG: <9> Feb 20 13:28:33 System: Stack unit 23 Temperature 58.0 C degrees, warning

SYSLOG: <12> Feb 20 13:28:33 System: Temperature is over warning level on unit 23
```

The following example uses the **show spx-mon pe-join** command to provide status of a PE unit joining through CB SPX port 2/1/1.

```
ICX7750-48F Router# show spx-mon pe-join 2/1/1
Error! Last PE 23 in the chain has no DOWNSTREAM SPX ports in UP state
Above error(s) needs to be corrected...
```

The following example uses the **show spx-mon history distributed** command to derive a history of ECP information directly from designated PE units as indicated by the keyword **distributed** in the command line. Without the keyword, ECP information is derived from the CB.

```
ICX7750-48F Router# show spx-mon history distributed 19

*********************
Response from PE 19:
*********************
51m17.8933 Unit 19 ecp_process_pdu: RxAck port_id 19/2/1, ecid 235, rxSeq 11, fSeq = 12, txSeq 12, fAck
0, ack 0, lSeq 11
   [stack: 002d1360 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.8786 Unit 19 ecp_process_pdu: RxAck port_id 19/2/1, ecid 235, rxSeq 10, fSeq = 11, txSeq 11, fAck
2, ack 2, lSeq 11
   [stack: 002d1360 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.8724 Unit 19 ecp_process_pdu: TxAck port_id 19/2/1, ecid 235, rxSeq 11, fSeq = 10, txSeq 11, fAck
1, ack 2, lSeq 11
   [stack: 002d201c 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.8667 Unit 19 ecp_transmit: port_id 19/2/1, ptype 1002, ecid 235, fSeq = 10, txSeq 11, fAck 1, ack
2, lSeq 11
   [stack: 002cd7e4 002f2bfc 002f53cc 002f6704 002ffa34 003008f8 002f7b2c 002f]
51m17.8393 Unit 19 ecp_process_pdu: RxPkt port_id 19/2/1, cc4e.2438.8e00, ecid 235, rxSeq 11, fSeq =
10, txSeq 10, fAck 1, ack 1, lSeq 10
   [stack: 002d18ec 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.8212 Unit 19 ecp_process_pdu: RxAck port_id 19/2/1, ecid 235, rxSeq 9, fSeq = 10, txSeq 10, fAck
1, ack 1, lSeq 10
   [stack: 002d1360 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.8148 Unit 19 ecp_process_pdu: TxAck port_id 19/2/1, ecid 235, rxSeq 10, fSeq = 9, txSeq 10, fAck
0, ack 1, lSeq 10
   [stack: 002d201c 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.8065 Unit 19 ecp_transmit: port_id 19/2/1, ptype 1002, ecid 235, fSeq = 9, txSeq 10, fAck 0, ack
1, lSeq 10
   [stack: 002cd7e4 002f2bfc 002f53cc 002f6704 002ffa34 003008f8 002f7b2c 002f]
51m17.7532 Unit 19 ecp_process_pdu: RxPkt port_id 19/2/1, cc4e.2438.8e00, ecid 235, rxSeq 10, fSeq = 9,
txSeq 9, fAck 0, ack 0, lSeq 9
   [stack: 002d18ec 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.6644 Unit 19 ecp_process_pdu: TxAck port_id 19/2/1, ecid 235, rxSeq 9, fSeq = 9, txSeq 9, fAck 0,
ack 0, lSeq 9
   [stack: 002d201c 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
51m17.6571 Unit 19 ecp_transmit: port_id 19/2/1, ptype 1002, ecid 235, fSeq = 9, txSeq 9, fAck 0, ack
0, lSeq 9
   [stack: 002cd7e4 002f2bfc 002f53cc 002f6704 002ffa34 003008f8 002f7b2c 002f]
51m17.6021 Unit 19 ecp_process_pdu: RxPkt port_id 19/2/1, cc4e.2438.8e00, ecid 235, rxSeq 9, fSeq = 9,
txSeq 9, fAck 0, ack 0, lSeq 8
   [stack: 002d18ec 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m53.2531 Unit 19 ecp_process_pdu: RxAck port_id 19/2/1, ecid 235, rxSeq 8, fSeq = 9, txSeq 9, fAck 0,
ack 0, lSeq 8
   [stack: 002d1360 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m53.1509 Unit 19 ecp_process_pdu: TxAck port_id 19/2/1, ecid 235, rxSeq 8, fSeq = 8, txSeq 8, fAck 0,
ack 0, lSeq 8
   [stack: 002d201c 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m53.1438 Unit 19 ecp_transmit: port_id 19/2/1, ptype 1002, ecid 235, fSeq = 8, txSeq 8, fAck 0, ack
0, lSeq 8
   [stack: 002cd7e4 002f2bfc 002f53cc 002f6704 002ffa34 003008f8 002f7b2c 002f]
48m53.0729 Unit 19 ecp_process_pdu: RxPkt port_id 19/2/1, cc4e.2438.8e00, ecid 235, rxSeq 8, fSeq = 8,
txSeq 8, fAck 0, ack 0, lSeq 7
   [stack: 002d18ec 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m48.9443 Unit 19 ecp_process_pdu: RxAck port_id 19/2/1, ecid 235, rxSeq 7, fSeq = 8, txSeq 8, fAck 0,
ack 0, lSeq 7
   [stack: 002d1360 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m48.6568 Unit 19 ecp_process_pdu: TxAck port_id 19/2/1, ecid 235, rxSeq 7, fSeq = 7, txSeq 7, fAck 0,
ack 0, lSeq 7
   [stack: 002d201c 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m48.6499 Unit 19 ecp_transmit: port_id 19/2/1, ptype 1002, ecid 235, fSeq = 7, txSeq 7, fAck 0, ack
0, lSeq 7
   [stack: 002cd7e4 002f2bfc 002f53cc 002f6704 002ffa34 003008f8 002f7b2c 002f]
48m47.4032 Unit 19 ecp_process_pdu: RxPkt port_id 19/2/1, cc4e.2438.8e00, ecid 235, rxSeq 7, fSeq = 7,
txSeq 7, fAck 0, ack 0, lSeq 6
   [stack: 002d18ec 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
```

```
48m47.0706 Unit 19 ecp_process_pdu: RxAck port_id 19/2/1, ecid 235, rxSeq 6, fSeq = 7, txSeq 7, fAck 0,
ack 0, lSeq 6
   [stack: 002d1360 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m47.0431 Unit 19 ecp_process_pdu: TxAck port_id 19/2/1, ecid 235, rxSeq 6, fSeq = 6, txSeq 6, fAck 0,
ack 0, lSeq 6
   [stack: 002d201c 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m47.0358 Unit 19 ecp_transmit: port_id 19/2/1, ptype 1002, ecid 235, fSeq = 6, txSeq 6, fAck 0, ack
0, lSeq 6
   [stack: 002cd7e4 002f2bfc 002f53cc 002f6704 002ffa34 003008f8 002f7b2c 002f]
48m47.0283 Unit 19 ecp_process_pdu: RxPkt port_id 19/2/1, cc4e.2438.8e00, ecid 235, rxSeq 6, fSeq = 6,
txSeq 6, fAck 0, ack 0, lSeq 5
   [stack: 002d18ec 00a1a0cc 00f0eee0 00f0fcd8 00c309f8 010a34e0 00615180 010a]
48m18.2364 Unit 19 PE joined
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.61 | The distributed parameter was added. |

# show stack

Displays information about the units in a stack and a representation of the stack topology.

## Syntax

**show stack** *num*

## Parameters

***num*** Displays information for the specified stack unit ID.

## Modes

Privileged EXEC mode

## Command Output

The **show stack** command displays the following information:

| Output field | Description |
|---|---|
| ID | Specifies the identification number of the stack unit. Each unit in the stack has a unique ID number. |
| Type | Specifies the type (model) of the stack unit. |
| Role | Specifies the role of the stack unit. The roles are controller, standby, or member. |
| Mac Address | Specifies the MAC address of the stack unit. The roles are controller, standby, or member. |
| Pri | Specifies the priority value assigned to the stack unit. The default value is 128. |
| State | Specifies whether the stack unit is local or remote. A unit with a State value of Local is the active controller. Units with a State value of Remote are either standby units or member units. |
| Comment | Indicates if the stack unit is ready (available). |
| Role history | Tracks up to six role changes per stack unit. The initial state is always displayed. Overflow is indicated by ellipses (...). Standby to member role changes are not displayed. Role history also does not display the transition after bootup of a non-active controller unit from member to standby. |

# Examples

The following example displays information about a stack with six stack trunks, including a representation of the stack topology.

```
device# show stack

T=21h22m31.3: alone: standalone, D: dynamic cfg, S: static, A=10, B=11, C=12
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7750-48XGF active  cc4e.246d.9e00 128 local   Ready
2  S ICX7750-48XGF standby cc4e.246d.8d80   0 remote  Ready
3  S ICX7750-48XGF member  cc4e.246d.9b00   0 remote  Ready
4  S ICX7750-48XGF member  cc4e.246d.9c80   0 remote  Ready
5  S ICX7750-20QXG member  cc4e.2439.2a80   0 remote  Ready
6  S ICX7750-20QXG member  cc4e.2439.3700   0 remote  Ready
7  S ICX7750-20QXG member  cc4e.2439.3880   0 remote  Ready
8  S ICX7750-20QXG member  cc4e.2439.2d00   0 remote  Ready
9  S ICX7750-48XGC member  cc4e.2439.1a00   0 remote  Ready
10 S ICX7750-48XGC member  cc4e.2439.1680   0 remote  Ready
11 S ICX7750-48XGC member  cc4e.2439.1d80   0 remote  Ready
12 S ICX7750-48XGC member  cc4e.2439.1280   0 remote  Ready

    active
     +---+        +---+        +---+        +---+        +---+        +---+
 -2/1| 1 |2/4--3/1| C |3/4==2/1| B |2/4==2/1| A |2/4--2/1| 9 |2/4--2/1| 8 |2/4=
|    +---+        +---+        +---+        +---+        +---+        +---+   |
|                                                                              |
|  standby                                                                     |
|    +---+        +---+        +---+        +---+        +---+        +---+   |
-2/4| 2 |2/1==2/4| 3 |2/1--2/4| 4 |2/1==2/4| 5 |2/1--2/4| 6 |2/1==2/4| 7 |2/1=
     +---+        +---+        +---+        +---+        +---+        +---+
Standby u2 - protocols ready, can failover
Current stack management MAC is cc4e.246d.9e00
```

The following example includes a role history for a three-unit stack.

```
ICX7750-20Q Router# show stack
T=29m36.7: alone: standalone, D: dynamic cfg, S: static
ID Type          Role    Mac Address    Pri State  Comment
1S ICX7750-20QXG standby 748e.f8f9.6300 128 remote Ready
2S ICX7750-20QXG member  cc4e.2438.7280   0 remote Ready
4S ICX7750-20QXG active  cc4e.2438.7500 128 local  Ready

active                 standby
    +---+        +---+        +---+
-2/1| 4 |2/4--2/1| 2 |2/4--2/1| 1 |2/4-
|   +---+        +---+        +---+   |
|                                     |
|-------------------------------------|

Standby u1 -protocols ready, can failover or manually switch over

Role history: N: standalone, A: active, S: standby, M: member

U1: N->A->S->M->S, U2: M->S->A->S->M, U4: M->S->A
```

The following two examples show information for an ICX 7650 stack that includes rear-module operating mode.

```
ICX7650-48ZP# show stack
T=5d19h28m49.5: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7650-48ZP  active  609c.9f52.2c96 128 local   Ready
2  S ICX7650-48ZP  standby 609c.9f52.2b22  10 remote  Ready

    active       standby
     +---+        +---+
  3/1| 1 |3/3==3/3| 2 |3/1
     +---+        +---+
Standby u2 - protocols ready, can failover
Current stack management MAC is 609c.9f52.2c96
Stack ports are operating at 40G.

ICX7650-48ZP# show stack
T=5d19h28m49.5: alone: standalone, D: dynamic cfg, S: static
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7650-48ZP  active  609c.9f52.2c96 128 local   Ready
2  S ICX7650-48ZP  standby 609c.9f52.2b22  10 remote  Ready

    active       standby
     +---+        +---+
  3/1| 1 |3/2-2/3 | 2 |3/1
     +---+        +---+
Standby u2 - protocols ready, can failover
Current stack management MAC is 609c.9f52.2c96
Stack ports are operating at 100G.
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include role history transitions for all stack units. |
| 08.0.70 | This command was modified to include operating mode information for ICX 7650 stacks. |

# show stack connection

Displays a representation of stack topology and a detailed connection report that contains information on connection errors or hardware failures.

## Syntax

**show stack connection**

## Modes

Privileged EXEC mode

# Examples

The following example displays a representation of a ring topology that has seven stack units and details on each of the trunk link connections.

```
device# show stack connection
Probing the topology. Please wait ...
device#
    active
    +---+        +---+        +---+        +---+        +---+        +---+
=2/1| 4 |2/6==2/6| 3 |2/1==2/1| 2 |2/6==2/6| 1 |2/1==2/1| 7 |2/6==2/6| 6 |2/1=
|   +---+        +---+        +---+        +---+        +---+        +---+   |
|                                                                             |
|                                                                      standby  |
|                                                                       +---+   |
-------------------------------------------------------------------------2/1| 5 |2/6=
                                                                        +---+

trunk probe results: 7 links
Link 1: u7 -- u1, num=5
1: 1/2/1 (T0) <---> 7/2/1 (T0)
2: 1/2/2 (T0) <---> 7/2/2 (T0)
3: 1/2/3 (T0) <---> 7/2/3 (T0)
4: 1/2/4 (T0) <---> 7/2/4 (T0)
5: 1/2/5 (T0) <---> 7/2/5 (T0)
Link 2: u2 -- u1, num=5
1: 1/2/6 (T1) <---> 2/2/6 (T1)
2: 1/2/7 (T1) <---> 2/2/7 (T1)
3: 1/2/8 (T1) <---> 2/2/8 (T1)
4: 1/2/9 (T1) <---> 2/2/9 (T1)
5: 1/2/10(T1) <---> 2/2/10(T1)
Link 3: u3 -- u2, num=5
1: 2/2/1 (T0) <---> 3/2/1 (T0)
2: 2/2/2 (T0) <---> 3/2/2 (T0)
3: 2/2/3 (T0) <---> 3/2/3 (T0)
4: 2/2/4 (T0) <---> 3/2/4 (T0)
5: 2/2/5 (T0) <---> 3/2/5 (T0)
Link 4: u4 -- u3, num=5
1: 3/2/6 (T1) <---> 4/2/6 (T1)
2: 3/2/7 (T1) <---> 4/2/7 (T1)
3: 3/2/8 (T1) <---> 4/2/8 (T1)

4: 3/2/9 (T1) <---> 4/2/9 (T1)
5: 3/2/10(T1) <---> 4/2/10(T1)
Link 5: u5 -- u4, num=5
1: 4/2/1 (T0) <---> 5/2/1 (T0)
2: 4/2/2 (T0) <---> 5/2/2 (T0)
3: 4/2/3 (T0) <---> 5/2/3 (T0)
4: 4/2/4 (T0) <---> 5/2/4 (T0)
5: 4/2/5 (T0) <---> 5/2/5 (T0)
Link 6: u6 -- u5, num=5
1: 5/2/6 (T1) <---> 6/2/1 (T0)
2: 5/2/7 (T1) <---> 6/2/2 (T0)
3: 5/2/8 (T1) <---> 6/2/3 (T0)
4: 5/2/9 (T1) <---> 6/2/4 (T0)
5: 5/2/10(T1) <---> 6/2/5 (T0)
Link 7: u7 -- u6, num=5
1: 6/2/6 (T1) <---> 7/2/6 (T1)
2: 6/2/7 (T1) <---> 7/2/7 (T1)
3: 6/2/8 (T1) <---> 7/2/8 (T1)
4: 6/2/9 (T1) <---> 7/2/9 (T1)
5: 6/2/10(T1) <---> 7/2/10(T1)
CPU to CPU packets are fine between 7 units.
```

# show stack detail

Displays information on all units in the stack, including the role, MAC address, priority, status, and stack connections for each stack unit.

## Syntax

**show stack detail**

## Modes

Privileged EXEC mode

## Command Output

The **show stack detail** command displays the following information:

| Output field | Description |
|---|---|
| ID | Specifies the identification number of the stack unit. Each unit in the stack has a unique ID number. |
| Type | Specifies the type (model) of the stack unit. |
| Role | Specifies the role of the stack unit. The roles are controller, standby, or member. |
| Mac Address | Specifies the MAC address of the stack unit. The roles are controller, standby, or member. |
| Pri | Specifies the priority value assigned to the stack unit. The default value is 128. |
| State | Specifies whether the stack unit is local or remote. A unit with a State value of Local is the active controller. Units with a State value of Remote are either standby units or member units. |
| Comment | Indicates if the stack unit is ready (available). |
| Unit # | Specifies the number assigned to the stack unit. Each unit in the stack has a unique unit number. (This is the same as the ID of the stack unit.) |
| Stack Port Status | Indicates whether the stack port is connected or disconnected. A port with the up status of up is connected to the stack, and a ports with the status of down (dn) is not connected to the stack. |
| Neighbors | Indicates units in the stack that are connected together. Each unit in the stack is connected to at least one other stack unit. |
| System uptime | Indicates the amount of time that the stack unit has been running since the last reset. The System uptime is listed for each unit in the stack. |

# Examples

The following example displays information on a full ICX 7450 stack containing 12 units, with six different models.

```
device# show stack detail

T=17h38m45.2: alone: standalone, D: dynamic cfg, S: static, A=10, B=11, C=12
ID   Type          Role    Mac Address    Pri State   Comment
1  S ICX7450-24G   active  cc4e.246c.ff80 128 local   Ready
2  S ICX7450-24G   standby cc4e.246d.02c8   0 remote  Ready
3  S ICX7450-24G   member  cc4e.246c.ffd0   0 remote  Ready
4  S ICX7450-24P   member  cc4e.246d.0520   0 remote  Ready
5  S ICX7450-48G   member  cc4e.246d.1c78   0 remote  Ready
6  S ICX7450-48G   member  cc4e.246d.1b78   0 remote  Ready
7  S ICX7450-48G   member  cc4e.246d.1df8   0 remote  Ready
8  S ICX7450-48P   member  cc4e.2489.8640   0 remote  Ready
9  S ICX7450-48GF  member  cc4e.246d.1478   0 remote  Ready
10 D ICX7450-24P   member  cc4e.246d.0638   0 remote  Ready
11 D ICX7450-24P   member  cc4e.246d.0778   0 remote  Ready
12 D ICX7450-48P   member  cc4e.246d.2938   0 remote  Ready

    active       standby
     +---+        +---+        +---+        +---+        +---+        +---+
  3/1| 1 |4/1--3/1| 2 |4/1--3/1| 3 |4/1--3/1| 4 |4/1--3/1| 5 |4/1--3/1| 6 |4/1-
     +---+        +---+        +---+        +---+        +---+        +---+   |
                                                                               |
                                                                               |
     +---+        +---+        +---+        +---+        +---+        +---+   |
     | C |3/1--4/1| B |3/1--4/1| A |3/1--4/1| 9 |3/1--4/1| 8 |3/1--4/1| 7 |3/1-
     +---+        +---+        +---+        +---+        +---+        +---+
Will assign standby in 53 sec due to all ready

Standby u2 - wait for standby assignment due to election
Current stack management MAC is cc4e.246c.ff80

Image-Auto-Copy is Enabled.

      Stack Port Status                  Neighbors
Unit# Stack-port1       Stack-port2      Stack-port1       Stack-port2
1     dn (1/3/1)        up (1/4/1)       none              U2 (2/3/1)
2     up (2/3/1)        up (2/4/1)       U1 (1/4/1)        U3 (3/3/1)
3     up (3/3/1)        up (3/4/1)       U2 (2/4/1)        U4 (4/3/1)
4     up (4/3/1)        up (4/4/1)       U3 (3/4/1)        U5 (5/3/1)
5     up (5/3/1)        up (5/4/1)       U4 (4/4/1)        U6 (6/3/1)
6     up (6/3/1)        up (6/4/1)       U5 (5/4/1)        U7 (7/3/1)
7     up (7/3/1)        up (7/4/1)       U6 (6/4/1)        U8 (8/3/1)
8     up (8/3/1)        up (8/4/1)       U7 (7/4/1)        U9 (9/3/1)
9     up (9/3/1)        up (9/4/1)       U8 (8/4/1)        U10 (10/3/1)
10    up (10/3/1)       up (10/4/1)      U9 (9/4/1)        U11 (11/3/1)
11    up (11/3/1)       up (11/4/1)      U10 (10/4/1)      U12 (12/3/1)
12    up (12/3/1)       none             U11 (11/4/1)      none

Unit# System uptime
1     17 hours 38 minutes 45 seconds
2     17 hours 38 minutes 43 seconds
3     17 hours 38 minutes 45 seconds
4     17 hours 38 minutes 44 seconds
5     17 hours 38 minutes 44 seconds
6     17 hours 38 minutes 44 seconds
7     17 hours 38 minutes 44 seconds
8     17 hours 38 minutes 45 seconds
9     17 hours 38 minutes 43 seconds
10    17 hours 32 minutes 24 seconds
11    1 minutes 9 seconds
12    1 minutes 9 seconds
ICX7450-24 Route
```

# show stack failover

Displays information about stack failover.

## Syntax

**show stack failover**

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **show stack failover** command to view information about rapid failover for the stack. This command displays if the standby is ready to takeover or not.

## Examples

The following example shows which unit is the current standby device and its status.

```
device# show stack failover

Current standby is unit 2. state=ready
Standby u2 - protocols ready, can failover
```

# show stack flash

Displays information about flash memory for stack members.

## Syntax

**show stack flash**

## Modes

Privileged EXEC mode

## Usage Guidelines

Use the **show stack flash** command to display information about flash memory for stack members.

## Command Output

The **show stack flash** command displays the following information:

| Output field | Description |
|---|---|
| ID | Specifies the identification number of the stack unit. Each unit in the stack has a unique ID number. |
| role | Specifies the role of the stack unit. The roles are controller, standby, or member. |
| priority | Specifies the priority value assigned to the stack unit. The default value is 128. |
| config | Indicates the port state (up or down) and identifies the port by number (stack-ID/slot/port). A port with the up status of up is connected to the stack, and a ports with the status of down (dn) is not connected to the stack. |
| The rest of the fields are used for debug purposes only. | |

## Examples

The following example display flash memory information..

```
device# show stack flash
There is no startup-config.old
Stack flash that was read in bootup:
ICX7450-24P, ID =4, role= active, pri=200, config=1, jumbo=X PPVLAN=X S2M=0 FIPS=X
stack p: [0]=4/2/1 [1]=4/2/6 default p: 4/2/1(5) 4/2/6(5), , , hash-chain=X vlan#=X
ve#=X stp#=X
active-chg=0
Current written stack flash:
ICX7450-24P, ID =4, role= active, pri=200, config=1, jumbo=X PPVLAN=X S2M=0 FIPS=X
stack p: [0]=4/2/1 [1]=4/2/6 default p: 4/2/1(5) 4/2/6(5), , , hash-chain=X vlan#=X
ve#=X stp#=X
```

# show stack link-sync

Displays the status of the link synchronization.

## Syntax

**show stack link-sync status**

## Parameters

**status** Displays link status information.

## Modes

Privileged EXEC mode

## Command Output

The **show stack link-sync status** command displays the following information:

| Output field | Description |
|---|---|
| STACKING_LINK_GLOBAL_CTRL messages (sent, received) | Number of global control messages sent and received. |
| STACKING_LINK_INDIVIDUAL_CTRL messages (sent, received) | Number of individual link control messages sent and received. |
| STACKING_LINK_STATUS messages (sent, received) | Number of link status control messages sent and received. |
| STACKING_POE_SCTRL messages (sent, received) | Number of Power over Ethernet (POE) control messages sent and received. |
| STACKING_POE_STATUS messages (sent, received) | Number of POE status messages sent and received. |
| global_ctrl_dest | Hexadecimal address of the global control destination. |
| individual_ctrl_dest | Hexadecimal address of the individual link control destination |
| status_dest | Number representing the destination status. |

## Examples

The following example shows link synchronization information.

```
device# show stack link-sync status
STACKING_LINK_GLOBAL_CTRL messages sent: 0, received: 0
STACKING_LINK_INDIVIDUAL_CTRL messages sent: 0, received: 0
STACKING_LINK_STATUS messages sent: 235, received: 225
STACKING_POE_SCTRL messages sent: 0, received: 0
STACKING_POE_STATUS messages sent: 0, received: 0
global_ctrl_dest: 0
individual_ctrl_dest: 0
status_dest: 2
```

# show stack neighbors

Displays information about stack member neighbors.

## Syntax

**show stack neighbors**

## Modes

Privileged EXEC mode

## Usage Guidelines

Stack neighbors are identified by unit ID for each stack unit.

## Command Output

The **show stack neighbors** command displays the following information:

| Output field | Description |
|---|---|
| U# | The identification number of the unit in the stack. Each unit in the stack has a unique identification number. |
| Stack-port1 | Identifies the neighbor stack unit for stack-port1 of the stack unit with this unit identification number (U#). The neighbor stack unit for stack-port1 of each unit in the stack is listed. |
| Stack-port2 | Identifies the neighbor stack unit for stack-port2 of the stack unit with this unit identification number (U#). The neighbor stack unit for stack-port2 of each unit in the stack is listed. |

## Examples

The following example output is for a device in a stack with three members.

```
device# show stack neighbors
U#   Stack-port1                Stack-port2
1   (1/2/1-1/2/2) to U2         (2/2/4-2/2/5) (1/2/4-1/2/6) to U3 (3/2/1-3/2/3)
2         (2/2/1) to U3 (3/2/4)               (2/2/4-2/2/5) to U1 (1/2/1-1/2/2)
3   (3/2/1-3/2/3) to U1               (1/2/4-1/2/6) (3/2/4) to U2 (2/2/1)
```

# show stack rel-ipc stats

Displays statistics on reliable Interprocessor Communications (IPC) communications that occur between stack units during a session.

## Syntax

**show stack rel-ipc stats** { **unit** *num* }

## Parameters

**rel-ipc**

Abbreviation for reliable Interprocessor Communications, which designates the proprietary packets exchanged between stack units during a communications session.

**stats**

Session statistics.

**unit** *num*

Optional parameter used to specify the stack unit number for which session statistics are to be displayed. If you do not specify a stack unit, session statistics are displayed for all units in the stack.

## Modes

Privileged EXEC mode

## Usage Guidelines

To display session statistics for a particular stack unit, specify the stack unit using the **unit** *num* parameters.

To display session statistics for all units in the stack, do not specify a stack unit.

## Command Output

Depending on whether you specify a stack unit, the **show stack rel-ipc stats** command displays reliable IPC statistics for all units in the stack, or for a single unit in the stack. See the example output below.

# Examples

The following example is reliable IPC statistics for a stack.

```
device# show stack rel-ipc stats unit 3
Unit 3 statistics:
Msgs: sent 907  recv 384, Pkt sends failed: 0, KA: sent 1522 recv 1522

Message types sent:
    [9]=846,          [13]=2,           [15]=31,          [59]=1,
    [76]=3,           [87]=24,

Message types received:
    [9]=366,          [13]=1,           [15]=17,

Session: base-channel, to U3, channel 0
buf size: xmt=4194312, rcv=65544, max msg=32776
State: established (last 19 minute(s) 16 second(s) ago) cnt: 1
Remote resets:                    0,      Reset packets sent:              0
Connection statistics (for current connection, if established):
Msgs sent:                      520,      Msgs rcvd:                     308
Atomic batches sent:              0,      Atomic batches rcvd:             0
Pkts sent:                     1325,      Pkts rcvd:                     945
Msg bytes sent:              262915,      Msg bytes rcvd:             131550
Pkt bytes sent:              631680,      Pkt bytes rcvd:             247560
Keepalive sent:                 231,      Keepalive rcvd:                231
Keepalive age:                    0,      Keepalive NBR age :              1
Flushes requested: 10, Suspends: 0, Resumes: 0
Packets sent with data (DAT), ACKs, and window updates (WND):
Other:      259,  ACK:          300,  WND:             6,  ACK+WND:         0
DAT:        753,  DAT+ACK:        7,  DAT+WND          0,  DA+AC+WND        0
Data retransmits done:          430,      Zero-window probes sent:         9
Dup ACK pkts rcvd:               40,      Pkts rcvd w/dup data:            0
Pkts rcvd w/data past window: 0

Session: ACL, to U3, channel 3
buf size: xmt=409608, rcv=131080, max msg=1472
State: established (last 19 minute(s) 16 second(s) ago) cnt: 1
Remote resets:                    0,      Reset packets sent:              0
Connection statistics (for current connection, if established):
Msgs sent:                      291,      Msgs rcvd:                      58
Atomic batches sent:              0,      Atomic batches rcvd:             0
Pkts sent:                      681,      Pkts rcvd:                     205
Msg bytes sent:              277656,      Msg bytes rcvd:              82128
Pkt bytes sent:              349288,      Pkt bytes rcvd:              84820
Keepalive sent:                 231,      Keepalive rcvd:                231
Keepalive age:                    0,      Keepalive NBR age :              1
Flushes requested: 0, Suspends: 0, Resumes: 0
Packets sent with data (DAT), ACKs, and window updates (WND):
Other:      232,  ACK:           12,  WND:             1,  ACK+WND:         2
DAT:        429,  DAT+ACK:        5,  DAT+WND          0,  DA+AC+WND        0
Data retransmits done:          272,      Zero-window probes sent:         5
Dup ACK pkts rcvd:               20,      Pkts rcvd w/dup data:            0
Pkts rcvd w/data past window: 0

Session: sync-reliable, to U3, channel 4
buf size: xmt=153608, rcv=10248, max msg=1472
State: established (last 16 minute(s) 38 second(s) ago) cnt: 1
Remote resets:                    0,      Reset packets sent:              0
Connection statistics (for current connection, if established):
Msgs sent:                       53,      Msgs rcvd:                       1
Atomic batches sent:              0,      Atomic batches rcvd:             0
Pkts sent:                      256,      Pkts rcvd:                      35
Msg bytes sent:               77380,      Msg bytes rcvd:               1460
Pkt bytes sent:              270984,      Pkt bytes rcvd:               1884
Keepalive sent:                 200,      Keepalive rcvd:                200
Keepalive age:                    0,      Keepalive NBR age :              1
Flushes requested: 0, Suspends: 0, Resumes: 0
Packets sent with data (DAT), ACKs, and window updates (WND):
Other:      201,  ACK:            1,  WND:             0,  ACK+WND:         0
DAT:         54,  DAT+ACK:        0,  DAT+WND          0,  DA+AC+WND        0
```

```
Data retransmits done:           41,      Zero-window probes sent:          0
Dup ACK pkts rcvd:               17,      Pkts rcvd w/dup data:             0
Pkts rcvd w/data past window: 0


Session: rcon-svr-to-3, to U3, channel 12
buf size: xmt=4008, rcv=8008, max msg=2668
State: established (last 19 minute(s) 14 second(s) ago) cnt: 1
Remote resets:                    0,      Reset packets sent:               0
Connection statistics (for current connection, if established):
Msgs sent:                       31,      Msgs rcvd:                       17
Atomic batches sent:              0,      Atomic batches rcvd:              0
Pkts sent:                      300,      Pkts rcvd:                       49
Msg bytes sent:                3592,      Msg bytes rcvd:                 155
Pkt bytes sent:               21836,      Pkt bytes rcvd:                 996
Keepalive sent:                 231,      Keepalive rcvd:                 231
Keepalive age:                    0,      Keepalive NBR age :               1
Flushes requested: 23, Suspends: 0, Resumes: 0
Packets sent with data (DAT), ACKs, and window updates (WND):
Other:      237,  ACK:           14,  WND:             0,  ACK+WND:         0
DAT:         49,  DAT+ACK:        0,  DAT+WND          0,  DA+AC+WND        0
Data retransmits done:           26,      Zero-window probes sent:          0
Dup ACK pkts rcvd:               10,      Pkts rcvd w/dup data:             0
Pkts rcvd w/data past window: 0
```

# show stack stack-ports

Displays status information about stack-ports.

## Syntax

**show stack stack-ports**

## Modes

Privileged EXEC mode

Global configuration mode

## Command Output

For ICX devices, an equal sign is used to indicate connections between trunk ports and the up port status is listed for all trunked ports. The **show stack stack-ports** command displays the following information:

| Output field | Description |
|---|---|
| U# or ID | Stack unit identification number. |
| Stack-port 1 | Indicates port status (up or down) and identifies the port by number (stack-ID/slot/port). |
| Stack-port 2 | Indicates port status (up or down) and identifies the port by number (stack-ID/slot/port). |
| Stack-ID up (stack-ID/slot/port) | Indicates status (up or down) for the stack unit and the status (up or down) of all configured stacking ports on the unit by number (stack-ID/slot/port). |

## Examples

The following output displays information about stack port status. Equal signs (=) in command output show connections between trunk ports.

```
device(config)# show stack stack-ports

    active       standby
     +---+        +---+        +---+
=2/1| 1 |2/4==2/1| 3 |2/4--2/1| 2 |2/4=
|   +---+        +---+        +---+   |
|                                     |
|-------------------------------------|

U#  Stack-port1                                 Stack-port2
1   up (1/2/1-1/2/2)                            up (1/2/4-1/2/6)
    up ports: 1/2/1
    up ports: 1/2/4

2   up (2/2/1)                                  up (2/2/4-2/2/5)
    up ports: 2/2/4

3   up (3/2/1-3/2/3)                            up (3/2/4)
    up ports: 3/2/1
```

# show startup-config (SPX)

Displays the startup configuration the PE unit would use in regular switch or router mode.

## Syntax

**show startup-config**

## Modes

PE mode

Provisional-PE mode

## Usage Guidelines

This command is available only on an ICX 7450 unit configured as an 802.1br Provisional-PE or PE unit with the **spx pe-enable** command. The **show startup-config** command shows the startup configuration that would be used by this unit if it were operating in regular mode as a switch or router.

In PE or Provisional-PE mode, the **show configuration** command shows the configuration in the PE startup file for this unit. In regular switch or router mode, use the **show running-config** command to show the currently running switch or router configuration.

# Examples

The following example displays the configuration this active PE would have if it returned to regular mode.

```
[PE]local-id@device# show startup-config
*** display startup configuration used in switch/router (not PE) ***
!
Startup-config data location is flash memory
!
Startup configuration:
!
ver 08.0.40b739T213
!
stack unit 1
  module 1 icx7450-24-port-management-module
  module 2 icx7400-xgc-4port-40g-module
  module 3 icx7400-qsfp-1port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
!
!
interface management 1
ip address 10.20.226.194 255.255.255.0
!
interface ethernet 1/2/1
speed-duplex 10G-full
!
interface ethernet 1/2/2
speed-duplex 10G-full
!
interface ethernet 1/2/3
speed-duplex 10G-full
!
interface ethernet 1/2/4
speed-duplex 10G-full
!
!
End
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# show statistics

Displays packet statistics.

## Syntax

**show statistics** [ **brief** ] [ **management** *num* | **unit** *unit-number* ]

**show statistics** [ **brief** ] [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ]

**show statistics** [ **brief** ] [ **lag** *lag-id* [ **to** *lag-id* | [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ] [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ]... ]

## Parameters

**brief**
Displays brief output.

**management** *num*
Displays packet statistics on the specified management interface.

**unit** *unit-number*
Displays packet statistics on all ports in a specific stack unit.

**ethernet** *unit/slot/port*
Displays packet statistics on a specific Ethernet interface.

**to** *unit/slot/port*
Displays packet statistics on a range of Ethernet interfaces.

**lag** *lag-id*
Specifies the LAG virtual interface.

**to** *lag-id*
Specifies a range of LAG virtual interface IDs.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

When you use the **brief** option, the output will have fewer fields.

You can view the packet statistics for a specific Ethernet interface, a list of Ethernet interfaces, and a range of Ethernet interfaces.

# Command Output

The **show statistics ethernet** and **show statistics management** command display the following information.

**NOTE**
The output of the **show statistics** command without any options, and the output of the **show statistics** command when using the **brief** option along with **ethernet**, **management**, or **unit** options display only the Port, In Packets, Out Packets, In Errors, and Out Errors fields.

| Output field | Description |
|---|---|
| Port | The port number. |
| Link | The link state. |
| State | The Spanning Tree Protocol (STP) state. |
| Dupl | The mode (full-duplex or half-duplex). |
| Speed | The port speed (10 Mbps, 100 Mbps, or 1000 Mbps). |
| Trunk | The trunk group number, if the port is a member of a trunk group. |
| Tag | Whether the port is a tagged member of a VLAN. |
| Pri | The QoS forwarding priority of the port (level0 to level7). |
| MAC | The MAC address of the port. |
| Name | The name of the port, if you assigned a name. |
| InOctets | The total number of good octets and bad octets received. |
| OutOctets | The total number of good octets and bad octets sent. |
| InPkts | The total number of packets received. The count includes rejected and local packets that are not sent to the switching core for transmission.<br><br>**NOTE**<br>In the output of the **show statistics** command without any options and when using the **brief** option along with the **ethernet**, **management**, or **unit** options, this field is shows as "In Packets." |
| OutPkts | The total number of good packets sent. The count includes unicast, multicast, and broadcast packets.<br><br>**NOTE**<br>In the output of the **show statistics** command without any options and when using the **brief** option along with the **ethernet**, **management**, or **unit** options, this field is shows as "Out Packets." |
| InBroadcastPkts | The total number of good broadcast packets received. |
| OutBroadcastPkts | The total number of good broadcast packets sent. |
| InMulticastPkts | The total number of good multicast packets received. |
| OutMulticastPkts | The total number of good multicast packets sent. |
| InUnicastPkts | The total number of good unicast packets received. |
| OutUnicastPkts | The total number of good unicast packets sent. |
| InBadPkts | The total number of packets received for which one of the following is true:<br>• The CRC was invalid.<br>• The packet was oversized.<br>• Jabbers: The packets were longer than 1518 octets and had a bad FCS.<br>• Fragments: The packets were less than 64 octets long and had a bad FCS.<br>• The packet was undersized (short). |

| Output field | Description |
|---|---|
| InFragments | The total number of packets received for which both of the following were true:<br>• The length was less than 64 bytes.<br>• The CRC was invalid. |
| InDiscards | The total number of packets that were received and then dropped due to a lack of receive buffers. |
| CRC | The total number of packets received for which all of the following was true:<br>• The data length was between 64 bytes and the maximum allowable frame size.<br>• No collision or late collision was detected.<br>• The CRC was invalid. |
| Collisions | The total number of packets received in which a collision event was detected. |
| LateCollisions | The total number of packets received in which a collision event was detected, but for which a receive error (Rx error) event was not detected. |
| InErrors | The total number of packets received that had alignment errors or physical errors. Excessive errors for some counters usually indicate a problem. When you operate at a half-duplex setting, some data link errors incrementing in Frame Check Sequence (FCS), alignment, runts, and collision counters are normal. Generally, a one percent ratio of errors to total traffic is acceptable for half-duplex connections. If the ratio of errors to input packets is greater than two or three percent, performance degradation can be noticed. In half-duplex environments, it is possible for both the switch and the connected device to sense the wire and transmit at exactly the same time, resulting in a collision. Collisions may cause runts, FCS, and alignment errors due to the frame not being completely copied to the wire, resulting in fragmented frames. When you operate at full-duplex, errors in FCS, Cyclic Redundancy Checks (CRC), alignment, and runt counters must be minimal.<br><br>**NOTE**<br>In the output of the **show statistics** command without any options and when using the **brief** option along with the **ethernet**, **management**, or **unit** options, this field is shown as "In Errors". |
| OutErrors | The total number of packets sent that had alignment errors or physical errors.<br><br>**NOTE**<br>In the output of the **show statistics** command without any options and when using the **brief** option along with the **ethernet**, **management**, or **unit** options, this field is shows as "Out Errors". |
| InGiantPkts | The total number of packets for which all of the following was true:<br>• The data length was longer than the maximum allowable frame size.<br>• No Rx error was detected.<br><br>**NOTE**<br>Packets are counted for this statistic regardless of whether the CRC is valid or invalid. |
| InShortPkts | The total number of packets received for which all of the following was true:<br>• The data length was less than 64 bytes.<br>• No Rx error was detected.<br>• No collision or late collision was detected.<br><br>**NOTE**<br>Packets are counted for this statistic regardless of whether the CRC is valid or invalid. |
| InJabber | The total number of packets received for which all of the following was true:<br>• The data length was longer than the maximum allowable frame size. |

| Output field | Description |
|---|---|
| | • No Rx error was detected.<br>• The CRC was invalid. |
| InFlowCtrlPkts | The total number of flow control packets received. |
| OutFlowCtrlPkts | The total number of flow control packets transmitted. |
| InBitsPerSec | The number of bits received per second. |
| OutBitsPerSec | The number of bits sent per second. |
| InPktsPerSec | The number of packets received per second. |
| OutPktsPerSec | The number of packets sent per second. |
| InUtilization | The percentage of the port bandwidth used by received traffic. |
| OutUtilization | The percentage of the port bandwidth used by sent traffic. |

# Examples

The following is sample output from the **show statistics brief management** command.

```
device(config)# show statistics brief management 1

Port      In Packets     Out Packets  Trunk     In Errors     Out Errors
mgmt1      39946         2             2       0             0
Total      39945         2             2          0             0
```

The following is sample output from the **show statistics management** command.

```
device# show statistics management 1

Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
mgmt1      Down    None    None None  None  No  None 0   748e.f80c.4100

 Port mgmt1 Counters:
        InOctets                    0          OutOctets                   0
          InPkts                    0            OutPkts                   0
  InBroadcastPkts                   0    OutBroadcastPkts                  0
  InMulticastPkts                   0    OutMulticastPkts                  0
    InUnicastPkts                   0      OutUnicastPkts                  0
        InBadPkts                   0
       InFragments                  0
        InDiscards                  0          OutErrors                   0
               CRC                  0         Collisions                   0
         InErrors                   0      LateCollisions                  0
       InGiantPkts                  0
       InShortPkts                  0
          InJabber                  0
    InFlowCtrlPkts                  0     OutFlowCtrlPkts                  0
      InBitsPerSec                  0       OutBitsPerSec                  0
      InPktsPerSec                  0       OutPktsPerSec                  0
     InUtilization              0.00%      OutUtilization              0.00%
```

The following is sample output from the **show statistics ethernet** command.

```
device# show statistics ethernet 1/1/1
Port       Link    State      Dupl Speed Trunk  Tag  Pvid Pri MAC             Name
1/1/1      Up      Forward    Half 100M  None   No   1    0   748e.f80c.4100

 Port 1/1/1 Counters:
        InOctets                3200          OutOctets                  256
          InPkts                  50            OutPkts                    4
  InBroadcastPkts                  0    OutBroadcastPkts                   3
  InMulticastPkts                 48    OutMulticastPkts                   0
    InUnicastPkts                  2      OutUnicastPkts                   1
        InBadPkts                  0
      InFragments                  0
       InDiscards                  0           OutErrors                   0
              CRC                  0          Collisions                   0
         InErrors                  0      LateCollisions                   0
      InGiantPkts                  0
      InShortPkts                  0
         InJabber                  0
   InFlowCtrlPkts                  0     OutFlowCtrlPkts                   0
     InBitsPerSec                264       OutBitsPerSec                  16
     InPktsPerSec                  0       OutPktsPerSec                   0
    InUtilization              0.00%      OutUtilization              0.00%
```

The following is sample output from the **show statistics brief** command.

```
device# show statistics brief

Port             In Packets       Out Packets       In Errors      Out Errors

1/1/1               7457812           7285553               3               0

1/1/2            5152995469              3731               3               0

1/1/3                472053         129827661               3               0

1/1/4            5153892037            441237               5               0

1/1/5                     0        4951785603               0               0

1/1/6                     0                 0               0               0

1/1/7                     0                 0               0               0

1/1/8                     0                 0               0               0

1/1/9                     0                 0               0               0

1/1/10                    0                 0               0               0

1/1/11                    0                 0               0               0

1/1/12                  829         138169869               0               0

lg1                     700                7000           0               0

lg256                   802                8002           0               0
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# show statistics dos-attack

Displays information about ICMP and TCP SYN packets dropped because burst thresholds were exceeded.

## Syntax

**show statistics dos-attack**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Examples

The following example displays output of the **show statistics dos-attack** command.

```
device# show statistics dos-attack
------------------------------ Local Attack Statistics -------------------------

                ICMP                                      TCP-SYN

---------------------------------------    ---------------------------------------

Dropped pkts Blocked pkts Lockup Count    Dropped pkts Blocked pkts Lockup Count

------------ ------------ ------------    ------------ ------------ ------------

0             0             0               0             0             0

----------------------------- Transit Attack Statistics ------------------------------

                      ICMP                                      TCP-SYN

-------------------------------------------    ---------------------------------------

Port/VE Dropped pkts Blocked pkts Lockup Count    Dropped pkts Blocked pkts Lockup Count

------- ------------ ------------ ------------    ------------ ------------ ------------

LG1      10           20           5000 111    600

IPv6 Address     LinkLayer-Addr  Age     Port/LAG     Virtual Port   vlan    VRF

 1212::11        f000.05b0.a78d  259198  lag lg2      lag lg2        12    default-
vrf


Total number of entries: 1
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | The command output was modified. |

# show statistics stack-ports

Displays information about all stacking ports in a stack topology.

## Syntax

**show statistics stack-ports**

## Modes

Privileged EXEC mode

## Command Output

The **show statistics stack-ports** command displays the following information:

| Output field | Description |
|---|---|
| Port | The number of the port (stack-unit number, slot number, and port number). |
| In Packets | The number of packets received on this port (incoming packets). |
| Out Packets | The number of packets sent from this port (outgoing packets). |
| In Errors | The number of errors received on this port (incoming errors). |
| Out Errors | The number of errors sent from this port (outgoing errors). |

## Examples

The following example output is statistics for all stack ports in a stack with seven member units.

```
device# show statistics stack-ports

Port      In Packets   Out Packets   In Errors   Out Errors
1/2/1          22223          4528           0            0
1/2/2          35506          3844           0            0
2/2/1           3161         34173           0            0
2/2/2          24721          3676           0            0
3/2/1           3048         23881           0            0
3/2/2          13540          2857           0            0
4/2/1           2862         13537           0            0
4/2/2           3626          3184           0            0
5/2/1           3183          3621           0            0
5/2/2           3265         13508           0            0
6/2/1          14020          3655           0            0
6/3/1           3652         17705           0            0
7/2/1          17705          3658           0            0
7/3/1           4047         21802           0            0
TOTAL         154559        153629           0            0
```

# show statistics traffic-policy

Displays the rate limiting traffic counters and the total packet count and byte count of the traffic filtered by ACL statements.

## Syntax

**show statistics traffic-policy** *TPD-name*

## Parameters

*TPD-name*

Specifies the name of the traffic policy definition for which you want to display ACL and traffic policy counters.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show statistics traffic-policy** command displays the following information.

| Output field | Description |
|---|---|
| Traffic Policy | The name of the traffic policy. |
| General Counters | |
| Port Region# | The port region to which the active traffic policy applies. |
| Byte Count | The number of bytes that were filtered (matched ACL clauses). |
| Packet Count | The number of packets that were filtered (matched ACL clauses). |
| Rate Limiting Counters (in bytes) | |
| Port Region# | The port region to which the active traffic policy applies. |
| Green Conformance | The number of bytes that did not exceed the committed information rate (CIR) packet rate. |
| Yellow Conformance | The number of bytes that exceeded the CIR packet rate. |
| Red Conformance | The number of bytes that exceeded the peak information rate (PIR) packet rate. |

# Examples

The following example shows sample output from the **show statistics traffic-policy** command. The output displays ACL and traffic policy counters.

```
device# show statistics traffic-policy tf125c

Traffic Policy tf125c:

General Counters:
Port Region#         Byte Count       Packet Count
---------------- -------------------- --------------------
0           235400192           1839051
All port regions  235400192           1839051

Rate Limiting Counters (in bytes):
Port Region#    Green/Yellow Conformance    Red Conformance
-------------- --------------------------- --------------------
0               225023872                   10376320
All port regs   225023872                   10376320
```

# show statistics tunnel

Displays statistical information for GRE and IPsec tunnels.

## Syntax

**show statistics tunnel** [ *tunnel-id* ]

## Parameters

*tunnel-id*

Specifies the tunnel ID. The default range is from 1 through 44. When the maximum number of GRE tunnels is set to 64 by using the **system-max gre-tunnels** command, the range is from 1 through 92.

## Modes

User EXEC mode

## Usage Guidelines

This command may be entered in all configuration modes.

## Command Output

The **show statistics tunnel** command displays the following information:

| Output field | | Description |
|---|---|---|
| Tunnel Status | | Indicates whether the tunnel and line protocol are up or down. Possible values are:<br>• up/up—The tunnel and line protocol are up.<br>• up/down—The tunnel is up and the line protocol is down.<br>• down/up—The tunnel is down and the line protocol is up.<br>• down/down—The tunnel and line protocol are down. |
| Packet Received | | Displays the number of packets received on the tunnel since it was last cleared by the administrator. |
| Packet Sent | | Displays the number of packets sent on the tunnel since it was last cleared by the administrator. |
| KA recv | | Displays the number of keepalive packets received on the tunnel since it was last cleared by the administrator. |
| KA sent | | Displays the number of keepalive packets sent on the tunnel since it was last cleared by the administrator. |
| GRE tunnel | | |
| | InOctets | Displays the number of incoming octets received on the tunnel since it was last cleared by the administrator. |
| | OutOctets | Displays the number of outgoing octets sent on the tunnel since it was last cleared by the administrator. |
| | InPkts | Displays the number of incoming packets received on the tunnel since it was last cleared by the administrator. |

| Output field | | Description |
|---|---|---|
| | OutPkts | Displays the number of outgoing packets sent on the tunnel since it was last cleared by the administrator. |
| IPsec tunnel | | |
| | Bytes Received | Displays the number of bytes received on the tunnel since it was last cleared by the administrator. |
| | Bytes Sent | Displays the number of bytes sent on the tunnel since it was last cleared by the administrator. |

# Examples

The following example shows how to display statistical information for tunnel 1.

```
device> show statistics tunnel 1

IP GRE Tunnels
      Tunnel Status   Packet Received   Packet Sent   KA recv   KA sent
  1   up/up           0                 0             0         0

IP  GRE Tunnel 1 HW Counters:
       InOctets            90             OutOctets             90
       InPkts               1             OutPkts                1
```

The following example shows how to display statistical information for all tunnels.

```
device> show statistics tunnel

IP GRE Tunnels
      Tunnel Status  Packet Received  Packet Sent  KA recv  KA sent

IPSEC Tunnels
      Tunnel Status  Packet Received  Packet Sent  Bytes Received  Bytes Sent
  9   down/down      0                0            0               0
  10  up/up          50               16442474     7300            9372173444
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was modified to display GRE-tunnel hardware counters on the Ruckus ICX 7000 series. |
| 8.0.50 | This command was modified to display IPsec tunnel information on the Ruckus ICX 7450. |

# show stp-bpdu-guard

Displays the BPDU guard state.

## Syntax

**show stp-bpdu-guard**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

## Examples

The following example displays the BPDU guard state.

```
device# show stp-bpdu-guard
BPDU Guard Enabled on:
Interface      Violation
Port 1/1/1     No
Port 1/1/2     No
Port 1/1/3     No
Port 1/1/4     No
Port 1/1/5     No
Port 1/1/6     No
Port 1/1/7     No
Port 1/1/8     No
Port 1/1/9     No
Port 1/1/10    No
Port 1/1/11    No
Port 1/1/12    Yes
Port 1/1/13    No
```

# show stp-group

Displays STP topology groups.

## Syntax

**show stp-group** [ *group-id* ]

## Parameters

*group-id*

Specifies the topology group ID.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

## Examples

The following example displays sample output of the **show stp-group** command.

```
device# show stp-group
Spanning tree Group 1
=================
 master-vlan 2
 member-vlan none

 Common control ports           L2 protocol
  no control ports configured
 Per vlan free ports
 ethernet 1/1/2                  Vlan 2
 ethernet 1/1/3                  Vlan 2
 ethernet 1/1/4                  Vlan 2
```

# show stp-protect-ports

Displays the STP protection configuration.

## Syntax

**show stp-protect-ports** [ **ethernet** *stackid/slot/port* ]

## Parameters

**ethernet** *stackid/slot/port*

Displays the STP protection configuration for a specific Ethernet interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

## Examples

The following example displays the STP protection configuration.

```
device# show stp-protect-ports
Port          BPDU Drop Count
1/1/3             478
1/1/5             213
1/1/6             0
1/1/12            31
```

The following example shows the STP protection configuration for a particular Ethernet interface.

```
device# show stp-protect-ports ethernet 1/1/3
STP-protect is enabled on port 1/1/3. BPDU drop count is 478
```

# show symmetric-flow-control

Displays the status of symmetric flow control as well as the default or configured total buffer limits and XON and XOFF thresholds.

## Syntax

**show symmetric-flow-control**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show symmetric-flow-control** command.

```
device# show symmetric-flow-control
Symmetric Flow Control Information:
-----------------------------------
SFC: Symmetric Flow Control
Defaults: 1G : Buffers: 272, XOFF Limit: 91, XON Limit: 75
          10G: Buffers: 416, XOFF Limit: 91, XON Limit: 75

     SFC           Total Buffers                    XOFF Limit                         XON Limit
Unit Enabled   1G      10G     1G          10G        1G          10G
---- -------   -------------  -------------------  -------------------
1    No        0       0      0(0%)      0(0%)      0(0%)       0(0%)      0(0%)
```

# show sz status

Displays the SmartZone IP address lists and information about the status of the connection with SmartZone.

## Syntax

**show sz status**

## Modes

Privileged EXEC mode

## Usage Guidelines

Beginning with SmartZone release 5.0, SmartZone can be used to monitor and manage ICX switches.

Use the **show sz status** command to display information about the ICX switch connection with SmartZone.

## Examples

The following example displays information about the connection with SmartZone:

```
ICX# show sz status

============    SZ Agent State Info     ===================
Config Status: None     Operation Status: Enabled
State: SZ SSH CONNECTED     Prev State: SZ SSH CONNECTING    Event: CONF

Active List         : 10.176.160.118
DHCP Option 43      : Yes
DHCP Opt 43 List    : 10.176.160.115
Passive List        : 10.176.160.118
Merged List         : 10.176.160.118, 10.176.160.115
Merged Idx: 0    IP : 10.176.160.118
Switch registrar host  : sw-registrar.ruckuswireless.com

SZ IP Used          : 10.176.160.118
SZ Query Status     :
        Response Received

SSH Tunnel Status - :
  Tunnel Status     : Established
  CLI IP/Port       : 127.255.255.253/53704
  SNMP IP/Port      : 127.255.255.254/53704
  Syslog IP/Port    : 127.0.0.1/20514

Timer Status        : Not Running
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |
| 08.0.80c | This command was modified to include the name of the switch registrar host. |

# show tech-support

Displays technical support information.

## Syntax

**show tech-support** [ acl | cluster | cpu | l2 | l3 { ipv4-uc | ipv6-uc } | license | memory | multicast | multicast6 | openflow | packet-loss | poe | stack ]

## Parameters

acl
: Displays ACL configuration log related details.

cluster
: Displays cluster related details.

cpu
: Displays CPU related details.

l2
: Displays Layer 2 related details.

l3
: Displays Layer 3 related details.

ipv4-uc
: Displays Layer 3 IPv4 elated details.

ipv6-uc
: Displays Layer 3 IPv6 elated details.

license
: Displays license elated details.

memory
: Displays memory related details.

multicast
: Displays multicast IPv4 related details.

multicast6
: Displays multicast IPv6 related details.

openflow
: Displays Openflow related details.

packet-loss
: Displays packet loss related details.

poe
: Displays a combination of output from multiple Power over Ethernet (PoE) technical support related commands including **show chassis**, **show inline power details**, **show inline power debug-info**, and **show inline power emesg**.

stack
Displays stack related details.

## Modes

Privileged exec mode

## Usage Guidelines

The **show tech support** commands can produce extensive output.

# Examples

To display technical support information for ACLs, use the following command.

```
device# show tech-support acl
================================================================================
BEGIN : show access-list all
CONTEXT : CONSOLE#0 : ACL CONFIG
TIME STAMP : 02:40:43.943 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

ACL Config Information.

Standard IP access list 10 : 0 entry

Standard IP access list 99 : 2 entry
10: permit host 10.0.0.0
20: permit 10.0.0.1 0.0.0.225


Extended IP access list 101 : 1 entry
10: permit ip host 10.0.0.2 any  802.1p-and-internal-marking 2


Extended IP access list 104 : 1 entry
10: permit ip host 10.0.0.2 any traffic-policy TPallow


Extended IP access list 105 : 1 entry
10: permit ip any any 802.1p-priority-matching 3 traffic-policy TPdrop


Extended IP access list 136 : 1 entry
10: permit ip any any 802.1p-priority-matching 3 traffic-policy adap

================================================================================
TIME STAMP : 02:40:43.944 GMT+00 Wed Jan 21 1970
END : show access-list all
TIME TAKEN : 238734 ticks (238734 nsec)
================================================================================
================================================================================
BEGIN : show acl-on-arp
CONTEXT : CONSOLE#0 : ARP ACL FILTERING
TIME STAMP : 02:40:43.944 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

ACL-ON-ARP list information
Port        ACL ID  Filter Count
================================================================================
================================================================================
TIME STAMP : 02:40:43.944 GMT+00 Wed Jan 21 1970
END : show acl-on-arp
TIME TAKEN : 47106 ticks (47106 nsec)
================================================================================
================================================================================
BEGIN : show access-list accounting
CONTEXT : CONSOLE#0 : ACL ACCOUNTING INFO
TIME STAMP : 02:40:43.968 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

ACL Accounting Information
Traffic Policy TPallow:

 General Counters:
 Port Region#               Byte Count          Packet Count
 ----------------  --------------------  --------------------
   0                                   0                     0
   All port regions                    0                     0
```

```
 Rate Limiting Counters (in bytes):
 Port Region#     Green/Yellow Conformance      Red Conformance
 --------------  --------------------------  --------------------
 0                                    0                    0
 All port regs                        0                    0
================================================================================
TIME STAMP : 02:40:43.968 GMT+00 Wed Jan 21 1970
END : show access-list accounting
TIME TAKEN : 48978 ticks (48978 nsec)
================================================================================
```

To display Layer 2 technical support information, use the following command.

```
ICX7450-24 Router#show tech-support l2
================================================================================
BEGIN : show version
CONTEXT : CONSOLE#0 : HW INFO
TIME STAMP : 02:44:34.943 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
  Copyright (c) 1996-2016 Ruckus Networks. All rights reserved.
    UNIT 1: compiled on Jun 23 2016 at 20:33:34 labeled as SPR08050b304
      (26308891 bytes) from Secondary SPR08050b304.bin
        SW: Version 08.0.50b304T213
      Compressed Boot-Monitor Image size = 786944, Version:10.1.08T215 (spz10108b004)
       Compiled on Wed Jun 15 11:56:14 2016

  HW: Stackable ICX7450-24
  Internal USB: Serial #: 9900614090900038
      Vendor: ATP Electronics, Total size = 1919 MB
================================================================================
UNIT 1: SL 1: ICX7450-24 24-port Management Module
      Serial  #:CYT3346K035
      License: ICX7450_L3_SOFT_PACKAGE   (LID: eavIIJLmFIK)
      License Compliance: ICX7450-PREM-LIC-SW is Compliant for next 45 days
      P-ASIC  0: type B548, rev 01  Chip BCM56548_A0
================================================================================
UNIT 1: SL 2: ICX7400-4X10GF 4-port 40G Module
      Serial  #:CYV3346K07G
================================================================================
UNIT 1: SL 3: ICX7400-1X40GQ 1-port 40G Module
      Serial  #:CYX3346K06F
================================================================================
UNIT 1: SL 4: ICX7400-1X40GQ 1-port 40G Module
      Serial  #:CYX3346K00A
================================================================================
 1000 MHz ARM processor ARMv7 88 MHz bus
 8192 KB boot flash memory
 2048 MB code flash memory
 2048 MB DRAM
STACKID 1  system uptime is 20 day(s) 2 hour(s) 44 minute(s) 1 second(s)
The system started at 00:00:55 GMT+00 Thu Jan 01 1970

The system : started=warm start   reloaded=by "reload"
*** NOT FOR PRODUCTION ***

================================================================================
TIME STAMP : 02:44:34.951 GMT+00 Wed Jan 21 1970
END : show version
TIME TAKEN : 4001646 ticks (4001646 nsec)
================================================================================

... (output truncated)

================================================================================
BEGIN : show port security
CONTEXT : CONSOLE#0 : PORT SECURITY INFO
TIME STAMP : 02:44:35.399 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

port security statistics
Unit/Module 1/1:
  Total ports: 0
  Total MAC address(es): 0
  Total violations: 0
  Total shutdown ports 0
Unit/Module 1/2:
  Total ports: 0
  Total MAC address(es): 0
  Total violations: 0
  Total shutdown ports 0
```

```
Unit/Module 1/3:
  Total ports: 0
  Total MAC address(es): 0
  Total violations: 0
  Total shutdown ports 0
Unit/Module 1/4:
  Total ports: 0
  Total MAC address(es): 0
  Total violations: 0
  Total shutdown ports 0
================================================================================
TIME STAMP : 02:44:35.402 GMT+00 Wed Jan 21 1970
END : show port security
TIME TAKEN : 1386332 ticks (1386332 nsec)
================================================================================
================================================================================
BEGIN : show metro-ring
CONTEXT : CONSOLE#0 : METRO RING INFO
TIME STAMP : 02:44:35.410 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

MRP Information :
Total MRP entries configured = 0
================================================================================
TIME STAMP : 02:44:35.410 GMT+00 Wed Jan 21 1970
END : show metro-ring
TIME TAKEN : 25518 ticks (25518 nsec)
================================================================================
================================================================================
BEGIN : show vsrp vrid <VRID>
CONTEXT : CONSOLE#0 : VSRP INFO
TIME STAMP : 02:44:35.410 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

VSRP Information:
router vsrp is not enabled
================================================================================
TIME STAMP : 02:44:35.410 GMT+00 Wed Jan 21 1970
END : show vsrp vrid <VRID>
TIME TAKEN : 13802 ticks (13802 nsec)
================================================================================
```

To display memory technical support information, use the following command.

```
device# show tech-support memory
================================================================================
BEGIN : show memory
CONTEXT : CONSOLE#0 : DRAM
TIME STAMP : 00:32:30.010 GMT+00 Thu Jan 01 1970
HW/SW INFO : ICX7450-24/SPR08050b347
================================================================================

MEMORY Related Information :
Stack unit 1:
Total DRAM: 2147483648 bytes
  Dynamic memory: 2095988736 bytes total, 1660276736 bytes free, 20% used

FLASH Related Information :
Stack unit 1:
  Compressed Pri Code size = 26386572, Version:08.0.50T213 (SPR08050b347.bin)
  Compressed Sec Code size = 26386572, Version:08.0.50T213 (SPR08050b347.bin)
  Compressed Boot-Monitor Image size = 786944, Version:10.1.08T215
  Code Flash Free Space = 1772818432
================================================================================
TIME STAMP : 00:32:30.062 GMT+00 Thu Jan 01 1970
END : show memory
TIME TAKEN : 25997977 ticks (25997977 nsec)
================================================================================
```

To display a combination of the output of multiple PoE-related commands, use the following command.

```
device# show tech-support poe

================================================================================
BEGIN : show running-config
CONTEXT : CONSOLE#0 : CONFIG
TIME STAMP : 00:56:34.371 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
Current configuration:
!
ver 08.0.50b304T213

... (output truncated)

!
end

================================================================================
TIME STAMP : 01:01:06.443 GMT+00 Wed Jan 21 1970
END : show running-config
TIME TAKEN : 62431402 ticks (62431402 nsec)
================================================================================
================================================================================
BEGIN : show version
CONTEXT : CONSOLE#0 : HW INFO
TIME STAMP : 01:01:06.443 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
  Copyright (c) 1996-2016 Ruckus Networks. All rights reserved.
    UNIT 1: compiled on Jun 23 2016 at 20:33:34 labeled as SPR08050b304
      (26308891 bytes) from Secondary SPR08050b304.bin
        SW: Version 08.0.50b304T213
      Compressed Boot-Monitor Image size = 786944, Version:10.1.08T215 (spz10108b004)
       Compiled on Wed Jun 15 11:56:14 2016

  HW: Stackable ICX7450-24
  Internal USB: Serial #: 9900614090900038
      Vendor: ATP Electronics, Total size = 1919 MB
================================================================================
UNIT 1: SL 1: ICX7450-24 24-port Management Module
      Serial  #:CYT3346K035
      License: ICX7450_L3_SOFT_PACKAGE   (LID: eavIIJLmFIK)
      License Compliance: ICX7450-PREM-LIC-SW is Compliant for next 45 days
      P-ASIC  0: type B548, rev 01  Chip BCM56548_A0
================================================================================
UNIT 1: SL 2: ICX7400-4X10GF 4-port 40G Module
      Serial  #:CYV3346K07G
================================================================================
UNIT 1: SL 3: ICX7400-1X40GQ 1-port 40G Module
      Serial  #:CYX3346K06F
================================================================================
UNIT 1: SL 4: ICX7400-1X40GQ 1-port 40G Module
      Serial  #:CYX3346K00A
================================================================================
 1000 MHz ARM processor ARMv7 88 MHz bus
 8192 KB boot flash memory
 2048 MB code flash memory
 2048 MB DRAM
STACKID 1  system uptime is 20 day(s) 1 hour(s) 33 second(s)
The system started at 00:00:55 GMT+00 Thu Jan 01 1970

The system : started=warm start   reloaded=by "reload"

================================================================================
TIME STAMP : 01:01:06.451 GMT+00 Wed Jan 21 1970
END : show version
TIME TAKEN : 3951305 ticks (3951305 nsec)
================================================================================
================================================================================
```

```
BEGIN : show interfaces brief
CONTEXT : CONSOLE#0 : PORT STATUS
TIME STAMP : 01:01:06.451 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/1      Down    None    None None  None  Yes 4000 0   cc4e.248b.b050  ERSPAN
1/1/2      Down    None    None None  None  No  5    0   cc4e.248b.b050
1/1/3      Down    None    None None  None  No  5    0   cc4e.248b.b052
1/1/4      Down    None    None None  None  No  5    0   cc4e.248b.b053
1/1/5      Down    None    None None  2     Yes N/A  0   cc4e.248b.b054
1/1/6      Down    None    None None  None  No  4000 0   cc4e.248b.b055
1/1/7      Down    None    None None  2     Yes N/A  0   cc4e.248b.b054
1/1/8      Down    None    None None  2     Yes N/A  0   cc4e.248b.b054
1/1/9      Down    None    None None  None  Yes N/A  0   cc4e.248b.b058
1/1/10     Down    None    None None  None  Yes N/A  0   cc4e.248b.b059
1/1/11     Down    None    None None  None  Yes N/A  0   cc4e.248b.b05a
1/1/12     Down    None    None None  None  Yes N/A  0   cc4e.248b.b05b
1/1/13     Down    None    None None  None  Yes N/A  0   cc4e.248b.b05c
1/1/14     Down    None    None None  None  Yes N/A  0   cc4e.248b.b05d
1/1/15     Down    None    None None  None  Yes N/A  0   cc4e.248b.b05e
1/1/16     Down    None    None None  None  Yes N/A  0   cc4e.248b.b05f
1/1/17     Down    None    None None  None  No  4000 0   cc4e.248b.b060
1/1/18     Down    None    None None  None  No  4000 0   cc4e.248b.b061
1/1/19     Down    None    None None  None  No  4000 0   cc4e.248b.b062
1/1/20     Down    None    None None  None  No  4000 0   cc4e.248b.b063
1/1/21     Down    None    None None  None  No  4000 0   cc4e.248b.b064
1/1/22     Down    None    None None  None  No  4000 0   cc4e.248b.b065
1/1/23     Down    None    None None  None  No  4000 0   cc4e.248b.b066
1/1/24     Down    None    None None  None  No  4000 0   cc4e.248b.b067
1/2/1      Down    None    None None  None  No  4000 0   cc4e.248b.b069
1/2/2      Down    None    None None  None  Yes N/A  0   cc4e.248b.b06a
1/2/3      Down    None    None None  None  No  4000 0   cc4e.248b.b06b
1/2/4      Down    None    None None  None  No  4000 0   cc4e.248b.b06c
1/3/1      Down    None    None None  None  No  4000 0   cc4e.248b.b050
1/4/1      Down    None    None None  None  Yes N/A  0   cc4e.248b.b050
mgmt1      Up      None    Full 1G    None  No  None 0   cc4e.248b.b050
ve10       Down    N/A     N/A  N/A   None  N/A N/A  N/A cc4e.248b.b050
ve100      Down    N/A     N/A  N/A   None  N/A N/A  N/A cc4e.248b.b050
lb1        Up      N/A     N/A  N/A   None  N/A N/A  N/A N/A
lb11       Up      N/A     N/A  N/A   None  N/A N/A  N/A N/A
tn1        Down    N/A     N/A  N/A   None  N/A N/A  N/A N/A
tn2        Down    N/A     N/A  N/A   None  N/A N/A  N/A N/A
tn8        Down    N/A     N/A  N/A   None  N/A N/A  N/A N/A
tn9        Down    N/A     N/A  N/A   None  N/A N/A  N/A N/A
tn11       Down    N/A     N/A  N/A   None  N/A N/A  N/A N/A
================================================================================
TIME STAMP : 01:01:06.502 GMT+00 Wed Jan 21 1970
END : show interfaces brief
TIME TAKEN : 25357514 ticks (25357514 nsec)
================================================================================
================================================================================
BEGIN : show statistics ethernet
CONTEXT : CONSOLE#0 : PACKET COUNTERS
TIME STAMP : 01:01:06.518 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
mgmt1      Up      None    Full 1G    None  No  None 0   cc4e.248b.b050

 Port mgmt1 Counters:
         InOctets          5333487552          OutOctets              4544
           InPkts              708739            OutPkts                71
  InBroadcastPkts              122793    OutBroadcastPkts                2
  InMulticastPkts              585944    OutMulticastPkts                0
    InUnicastPkts                   2      OutUnicastPkts               69
        InBadPkts                   0
      InFragments                   0
       InDiscards                   0           OutErrors                0
              CRC                   0         Collisions                 0
         InErrors                   0      LateCollisions                0
      InGiantPkts                   0
```

```
      InShortPkts                    0
         InJabber                    0         OutDiscards                    0
   InFlowCtrlPkts                    0     OutFlowCtrlPkts                    0
     InBitsPerSec                 5112       OutBitsPerSec                    0
     InPktsPerSec                    0       OutPktsPerSec                    0
    InUtilization                0.00%      OutUtilization                0.00%
    InPFCPkts [0]                    0      OutPFCPkts [0]                    0
    InPFCPkts [1]                    0      OutPFCPkts [1]                    0
    InPFCPkts [2]                    0      OutPFCPkts [2]                    0
    InPFCPkts [3]                    0      OutPFCPkts [3]                    0
    InPFCPkts [4]                    0      OutPFCPkts [4]                    0
    InPFCPkts [5]                    0      OutPFCPkts [5]                    0
    InPFCPkts [6]                    0      OutPFCPkts [6]                    0
    InPFCPkts [7]                    0      OutPFCPkts [7]                    0

================================================================================
TIME STAMP : 01:01:06.693 GMT+00 Wed Jan 21 1970
END : show statistics ethernet
TIME TAKEN : 87628812 ticks (87628812 nsec)
================================================================================
================================================================================
BEGIN : show logging
CONTEXT : CONSOLE#0 : STATIC/DYNAMIC LOG
TIME STAMP : 01:01:06.694 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
Syslog logging: enabled ( 0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 15 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
Jan  1 00:00:58:I:System: Stack unit 1   Power supply 2  is up

Dynamic Log Buffer (50 lines):
Jan 20 08:57:51:I:Security: console login by un-authenticated console user to PRIVILEGED EXEC mode
Jan 20 08:01:56:I:Security: console logout by un-authenticated console user from PRIVILEGED EXEC mode
Jan 20 06:32:05:I:Security: console login by un-authenticated console user to PRIVILEGED EXEC mode
Jan 19 23:27:53:I:Security: console logout by un-authenticated console user from PRIVILEGED EXEC mode
Jan 19 23:19:48:I:Security: running-config was changed by operator from console
Jan  7 07:13:20:I:Security: console login by un-authenticated console user to PRIVILEGED EXEC mode
Jan  5 10:46:17:I:Security: console logout by un-authenticated console user from PRIVILEGED EXEC mode
Jan  5 05:41:12:I:Security: running-config was changed by operator from console
Jan  4 03:31:38:I:Security: running-config was changed by operator from console
Jan  1 00:02:10:I:Security: console login by un-authenticated console user to PRIVILEGED EXEC mode
Jan  1 00:00:58:I:System: Stack unit 1   Power supply 2  is up
Jan  1 00:00:57:I:System: Interface ethernet mgmt1, state up
Jan  1 00:00:57:I:System: Warm start
Jan  1 00:00:57:I:DHCPC: protocol disabled
Jan  1 00:00:56:N:VRRP-Extended: intf state changed, intf v10, vrid 1, state initialized
================================================================================
TIME STAMP : 01:01:06.731 GMT+00 Wed Jan 21 1970
END : show logging
TIME TAKEN : 18858269 ticks (18858269 nsec)
================================================================================
================================================================================
BEGIN : show media
CONTEXT : CONSOLE#0 : OPTICS TYPE
TIME STAMP : 01:01:06.740 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
Port 1/1/1:   Type  : 1G M-C (Gig-Copper)
Port 1/1/2:   Type  : 1G M-C (Gig-Copper)
Port 1/1/3:   Type  : 1G M-C (Gig-Copper)
Port 1/1/4:   Type  : 1G M-C (Gig-Copper)
Port 1/1/5:   Type  : 1G M-C (Gig-Copper)
Port 1/1/6:   Type  : 1G M-C (Gig-Copper)
Port 1/1/7:   Type  : 1G M-C (Gig-Copper)
Port 1/1/8:   Type  : 1G M-C (Gig-Copper)
Port 1/1/9:   Type  : 1G M-C (Gig-Copper)
Port 1/1/10:  Type  : 1G M-C (Gig-Copper)
Port 1/1/11:  Type  : 1G M-C (Gig-Copper)
```

```
Port 1/1/12:  Type  : 1G M-C (Gig-Copper)
Port 1/1/13:  Type  : 1G M-C (Gig-Copper)
Port 1/1/14:  Type  : 1G M-C (Gig-Copper)
Port 1/1/15:  Type  : 1G M-C (Gig-Copper)
Port 1/1/16:  Type  : 1G M-C (Gig-Copper)
Port 1/1/17:  Type  : 1G M-C (Gig-Copper)
Port 1/1/18:  Type  : 1G M-C (Gig-Copper)
Port 1/1/19:  Type  : 1G M-C (Gig-Copper)
Port 1/1/20:  Type  : 1G M-C (Gig-Copper)
Port 1/1/21:  Type  : 1G M-C (Gig-Copper)
Port 1/1/22:  Type  : 1G M-C (Gig-Copper)
Port 1/1/23:  Type  : 1G M-C (Gig-Copper)
Port 1/1/24:  Type  : 1G M-C (Gig-Copper)
Port 1/2/1:   Type  : EMPTY
Port 1/2/2:   Type  : EMPTY
Port 1/2/3:   Type  : EMPTY
Port 1/2/4:   Type  : EMPTY
Port 1/3/1:   Type  : EMPTY
Port 1/4/1:   Type  : EMPTY
================================================================================
TIME STAMP : 01:01:06.743 GMT+00 Wed Jan 21 1970
END : show media
TIME TAKEN : 1498916 ticks (1498916 nsec)
================================================================================
================================================================================
BEGIN : show fdp neighbors
CONTEXT : CONSOLE#0 : FDP NEIGHBORS ENTRIES
TIME STAMP : 01:01:06.743 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
Neither FDP nor CDP is enabled
================================================================================
TIME STAMP : 01:01:06.743 GMT+00 Wed Jan 21 1970
END : show fdp neighbors
TIME TAKEN : 23955 ticks (23955 nsec)
================================================================================
================================================================================
BEGIN : show lldp neighbors
CONTEXT : CONSOLE#0 : LLDP NEIGHBORS ENTRIES
TIME STAMP : 01:01:06.743 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================
LLDP is not running
================================================================================
TIME STAMP : 01:01:06.743 GMT+00 Wed Jan 21 1970
END : show lldp neighbors
TIME TAKEN : 13672 ticks (13672 nsec)
================================================================================
================================================================================
BEGIN : show trunk
CONTEXT : CONSOLE#0 : TRUNK INFO
TIME STAMP : 01:01:06.744 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
================================================================================

Trunk Status Information :

Configured trunks:

Trunk ID: 3
Hw Trunk ID: 1
Ports_Configured: 3

Ports   PortName Port_Status Monitor Rx_Mirr Tx_Mirr Monitor_Dir
1/1/5   none     enable      off     N/A     N/A     N/A
1/1/7   none     enable      off     N/A     N/A     N/A
1/1/8   none     enable      off     N/A     N/A     N/A

Operational trunks:

Trunk ID: 3
Hw Trunk ID: 1
```

```
Duplex: None
Speed: None
Tag: Yes
Priority: level0
Active Ports: 0

==========================================================================
TIME STAMP : 01:01:06.745 GMT+00 Wed Jan 21 1970
END : show trunk
TIME TAKEN : 578548 ticks (578548 nsec)
==========================================================================
Ports   Link_Status port_state
1/1/5   down        Blocked
1/1/7   down        Blocked
1/1/8   down        Blocked

==========================================================================
BEGIN : show lag
CONTEXT : CONSOLE#0 : LAG INFO
TIME STAMP : 01:01:06.790 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
==========================================================================
Total number of LAGs:          2
Total number of deployed LAGs: 1
Total number of trunks created:1 (255 available)
LACP System Priority / ID:     1 / cc4e.248b.b050
LACP Long timeout:             120, default: 120
LACP Short timeout:            3, default: 3

=== LAG "lag1" ID 2 (static Deployed) ===
LAG Configuration:
   Ports:         e 1/1/5 e 1/1/7 to 1/1/8
   Port Count:    3
   Lag Interface:  lg2
   Trunk Type:    hash-based
Deployment: HW Trunk ID 1
Port       Link    State   Dupl Speed Trunk Tag Pvid Pri MAC             Name
1/1/5      Down    None    None None  2     Yes N/A  0   cc4e.248b.b054
1/1/7      Down    None    None None  2     Yes N/A  0   cc4e.248b.b054
1/1/8      Down    None    None None  2     Yes N/A  0   cc4e.248b.b054

=== LAG "pink" ID 1 (dynamic Not Deployed) ===
LAG Configuration:
   Ports:
   Port Count:    0
   Lag Interface:  lg1
   Trunk Type:    hash-based
   LACP Key:      20001


==========================================================================
TIME STAMP : 01:01:06.791 GMT+00 Wed Jan 21 1970
END : show running-config
TIME TAKEN : 362961 ticks (362961 nsec)
==========================================================================
==========================================================================
TIME STAMP : 01:01:06.791 GMT+00 Wed Jan 21 1970
END : show lag
TIME TAKEN : 404487 ticks (404487 nsec)
==========================================================================
==========================================================================
BEGIN : show poe
CONTEXT : CONSOLE#0 : poe INFO
TIME STAMP : 01:01:06.791 GMT+00 Wed Jan 21 1970
HW/SW INFO : ICX7450-24/SPR08050b304
==========================================================================

Log Size: 2000 entries.         Number of entries in use: 1.
Logging is active.
Log printing is requested for complete log.
+----------------+-----+-----+----------+-------------------------------------------+
|Timestamp       | Sys | Dev | Port     | Event Trace Message                       |
```

```
+----------------+-----+-----+----------+-------------------------------------------+
 Jan 20 23:56:24 |  N  | N/A | N/A      | PoE Event Log Mgr: User Req Logging On
+----------------+-----+-----+----------+-------------------------------------------+
==========================================================================
TIME STAMP : 01:01:06.791 GMT+00 Wed Jan 21 1970
END : show poe
TIME TAKEN : 54174 ticks (54174 nsec)
==========================================================================
```

# History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# show telnet

Displays Telnet connection and configuration details.

## Syntax

**show telnet** [ **config** ]

## Parameters

**config**

Displays Telnet configuration information.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show telnet config** command displays the following information:

| Output field | Description |
|---|---|
| Telnet Server | Telnet server status - enabled or disabled. |
| Idle timeout | The configured idle timeout of the Telnet server. |
| Login timeout | The configured login timeout of the Telnet server. |
| Login retries | The configured number of retries allowed to connect to the Telnet server. |
| Strict management VRF | Strict management VRF is enabled or disabled for the Telnet server. |
| Authentication | The authentication is enabled or disabled for the Telnet server. |
| suppress-reject-message | Whether the connection rejection message is suppressed or not; if a Ruckus device denies Telnet management access to the device, the software sends a message to the denied Telnet client. |

# Examples

The following example displays output of the **show telnet** command showing the Telnet connections and their status.

```
device(config)# show telnet
Console connections (by unit number):
1         established
                you are connecting to this session
                1 minutes 5 seconds in idle
2         established
                1 hours 4 minutes 18 seconds in idle
3         established
                1 hours 4 minutes 15 seconds in idle
4         established
                1 hours 4 minutes 9 seconds in idle
Telnet connections (inbound):
1         closed
2         closed
3         closed
4         closed
5         closed
Telnet connection (outbound):
6         closed
SSH connections:
1         closed
2         closed
3         closed
4         closed
5         closed
```

The following example displays output of the **show telnet config** command showing Telnet configuration details.

```
device(config)# show telnet config
Telnet server                   : Enabled
Idle timeout (minutes)          : 0
Login timeout (minutes)         : 2
Login retries                   : 4
Strict management VRF           : Disabled
Authentication                  : Disabled
suppress-reject-message         : Disabled
Telnet IPv4 clients             : All
Telnet IPv6 clients             : All
Telnet IPv4 access-group        :
Telnet IPv6 access-group        :
```

# show topology-group

Displays topology group information.

## Syntax

**show topology-group** [ *group-id* ]

## Parameters

*group-id*

Displays the information of the topology group of the specified ID.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VLAN configuration mode

## Usage Guidelines

## Command Output

The **show topology-group** command displays the following information:

| Output field | Description |
|---|---|
| master-vlan | The master VLAN for the topology group. The settings for STP, MRP, or VSRP on the control ports in the master VLAN apply to all control ports in the member VLANs within the topology group. |
| member-vlan | The member VLANs in the topology group. |
| Common control ports | The master VLAN ports that are configured with Layer 2 protocol information. The Layer 2 protocol configuration and state of these ports in the master VLAN applies to the same port numbers in all the member VLANs. |
| Per vlan free ports | The ports that are not controlled by the Layer 2 protocol information in the master VLAN. |

# Examples

The following example displays the topology group information.

```
device# show topology-group
Topology Group 3
================
master-vlan 2
member-vlan none
Common control ports       L2 protocol
ethernet 1/1/1             MRP
ethernet 1/1/2             MRP
ethernet 1/1/5             VSRP
ethernet 1/2/22            VSRP
Per vlan free ports
ethernet 1/2/3             Vlan 2
ethernet 1/1/4             Vlan 2
ethernet 1/2/11            Vlan 2
ethernet 1/2/12            Vlan 2
```

# History

| Release version | Command history |
|---|---|
| 16r.1.00 | This command was introduced. |

# show traffic-policy

Displays traffic policies that are currently defined on the device.

## Syntax

**show traffic-policy** [ *TPD-name* ]

## Parameters

*TPD-name*

Specifies the name of the traffic policy.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Command Output

The **show traffic-policy** command displays the following information.

| Output field | Description |
|---|---|
| Traffic Policy | The name of the traffic policy. |
| Metering | Shows whether rate limiting is configured as part of the traffic policy:<br>• Enabled: The traffic policy includes a rate-limiting configuration.<br>• Disabled: The traffic policy does not include a rate-limiting configuration. |
| Mode | If rate limiting is enabled, this field shows the type of metering enabled on the port:<br>• Fixed Rate-Limiting<br>• Adaptive Rate-Limiting |
| cir | The committed information rate, in kilobits per second, for the adaptive rate-limiting policy. |
| cbs | The committed burst size, in bytes, for the adaptive rate-limiting policy. |
| pir | The peak information rate, in kilobits per second, for the adaptive rate-limiting policy. |
| pbs | The peak burst size, in bytes, for the adaptive rate-limiting policy. |
| Counting | Shows whether ACL counting is configured as part of the traffic policy:<br>• Enabled: The traffic policy includes an ACL counting configuration.<br>• Not Enabled: The traffic policy does not include an ACL counting configuration. |
| Number of References/Bindings | The number of port regions to which this traffic policy applies. |

# Examples

The following example is sample output from the **show traffic-policy** command. The output displays traffic policies that are currently defined on the device.

```
device# show traffic-policy t_voip

Traffic Policy - t_voip:
Metering Enabled, Parameters:
Mode: Adaptive Rate-Limiting
cir: 100 Pkts/s, cbs: 2000 Pkts, pir: 200 Pkts/s, pbs: 4000 Pkts
Counting Not Enabled
```

# show transmit-counter

Displays traffic counter (also called transmit counters) profiles and traffic counter statistics.

## Syntax

**show transmit-counter** { **profiles** | **values** *number* }

## Parameters

**profiles**

Displays details of the traffic traffic counter profiles.

**values** *number*

Displays details of traffic queue counters. The number specifies a valid enhanced traffic counter in the range from 1 through 48.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Usage Guidelines

**NOTE**
Once the enhanced traffic counters are displayed, the counters are cleared (reset to zero).

## Command Output

The **show transmit-counter values** command displays the following information:

| Output field | Description |
|---|---|
| Transmitted Frames | The number of frames transmitted. |
| Known Unicast | The number of known unicast packets transmitted. |
| Multicast & Unknown Unicast | The number of multicast and unknown unicast packets transmitted. |
| Broadcast | The number of broadcast packets transmitted. |
| Dropped Frames | The number of dropped frames. |
| Bridge Egress Filtered | The number of bridged outbound packets that were filtered and dropped. This number includes the number of packets that were dropped because of any of the following conditions:<br>• The port was disabled or the link was down.<br>• The port or port region does not belong to the VLAN specified in the transmit counter configuration.<br>• A Layer 2 protocol (for example, spanning tree) had the port in a blocked state. |

| Output field | Description |
|---|---|
| | • The source port was suppressed for multi-target packets.<br>• The priority queue specified in the traffic counter was not allowed for some other reason.<br>• Unknown unicast and unregistered multicast packets were filtered. |
| Congestion Drops | The number of outbound packets that were dropped because of traffic congestion. |

# Examples

The following is a sample output of the **show transmit-counter profiles** command.

```
device# show transmit-counter profiles

Tx Counter     Port(s)               Vlan Id     Priority      Device        Set
1                 1/1/1-1/1/12       All         All              Dev 0         Set0
4                    1/ 1/18          1          7                        Dev 1          Set0
10                1/1/13-1/1/24   100          All              Dev 1         Set10
```

The following is sample output from the **show transmit-counter values** command.

```
device# show transmit-counter values 1

Transmit Queue Counter Values for Counter 1:
Transmitted Frames:
 Known Unicast               : 17204
 Multicast & Unknown Unicast : 2797
 Broadcast                   : 5
Dropped Frames:
 Bridge Egress Filtered      : 100
 Congestion Drops            : 0
```

# show users

Displays the user account information.

## Syntax

**show users**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Command Output

The **show users** command displays the following information:

| Output field | Description |
|---|---|
| Username | The username of each user. |
| Password | The password for each user. |
| Encrypt | Whether the password encryption is enabled or not. |
| Priv | The privilege level for the user: 0 - Super User level (full read-write access), 4 - Port Configuration level, 5 - Read Only level |
| Status | Whether the user status is enabled or not. |
| Expire Time | The password expiration time in days. |

## Examples

The following example displays output of the **show users** command.

```
device(config)# show users
Username   Password                              Encrypt   Priv Status   Expire Time
=============================================================================
wonka      $1$JVbXZqTW$g9N1/WUipXg6jM6OUKHQZ. enabled   0    enabled  Never
xyz        $1$13zNygo2$vXOKCwghNvXT/YegDawpU0 enabled   0    enabled  Never
aopo       $1$d04FqfAw$W6WiSw6gGJv//C1pvJFpQ. enabled   0    enabled  Never
```

# show version

This CLI displays information regarding software version running on each unit as well as the current boot-monitor version.

## Syntax

**show version**

## Modes

Privileged EXEC mode

## Usage Guidelines

Depending on device support, the serial numbers of the pluggable or fixed modules are displayed in the output. The role of the stack unit and its bootup ID are displayed in the last line of command output. No role is displayed for standalone units. Show version also displays the recommended u-boot version, even while displaying details for remote units and PEs.

Whenever current boot-version is not same as the recommended boot-monitor version, it will display alert message for the user to indicate the mismatch in boot-monitor version.

Note: Current boot-monitor version may be older or newer to the recommended version, to receive the alert message.

Similar information is displayed for the show version unit <unit num> CLI. This information will be displayed for each unit in standalone as well stacking or spx environment.

# Examples

The following is an example of the output displayed from the **show version** command, when run on an ICX 7450.

```
device#show version
Copyright (c) 2017 Ruckus Wireless, Inc. All rights reserved.
UNIT 2: compiled on Jan 8 2018 at 04:54:07 labeled as TNR08070a
(55796108 bytes) from Primary TNR08070a.bin
SW: Version 08.0.70aT233
Compressed Boot-Monitor Image size = 1573376, Version:10.1.11T235 (tnu10111)
Compiled on Wed Dec 13 11:14:03 2017
UNIT 1: compiled on Jan 8 2018 at 04:54:07 labeled as TNR08070a
(55796108 bytes) from Primary TNR08070a.bin
SW: Version 08.0.70aT233
Compressed Boot-Monitor Image size = 1573376, Version:10.1.11T235 (tnu10111)
UNIT 3: compiled on Jan 8 2018 at 04:54:07 labeled as TNR08070a
(55796108 bytes) from Primary TNR08070a.bin
SW: Version 08.0.70aT233
Compressed Boot-Monitor Image size = 1573376, Version:10.1.11T235 (tnu10111)
UNIT 4: compiled on Jan 8 2018 at 04:54:07 labeled as TNR08070a
(55796108 bytes) from Primary TNR08070a.bin
SW: Version 08.0.70aT233
Compressed Boot-Monitor Image size = 1573376, Version:10.1.11T235 (tnu10111)
HW: Stackable ICX7650-48Z-HPOE
==========================================================================
UNIT 1: SL 1: ICX7650-48F-L3-BASE 48-port Management Module
Serial #:EZE3324N01E
License: ICX7650_L3_SOFT_PACKAGE (LID: gbgIIHJpFGg)
==========================================================================
UNIT 1: SL 2: ICX7600-2X40GQ 2-port 80G Module
Serial #:EZG3320N04Z
==========================================================================
UNIT 1: SL 3: ICX7650-2X100G 2-port 200G Module
==========================================================================
UNIT 2: SL 1: ICX7650-48ZP-L3-BASE POE 48-port Management Module
Serial #:EZC3322N03X
License: ICX7650_L3_SOFT_PACKAGE (LID: gbeIIHHpFIz)
P-ASIC 2: type B568, rev 11 Chip BCM56568_B0
==========================================================================
UNIT 2: SL 2: ICX7600-2X40GQ 2-port 80G Module
Serial #:EZG3320N01J
==========================================================================
UNIT 2: SL 3: ICX7650-2X100G 2-port 200G Module
==========================================================================
UNIT 3: SL 1: ICX7650-48F-L3-BASE 48-port Management Module
Serial #:EZE3324N02V
License: ICX7650_L3_SOFT_PACKAGE (LID: gbgIIHJpFHx)
==========================================================================
UNIT 3: SL 2: ICX7600-1X100G 1-port 100G Module
Serial #:EZH3335N00H
==========================================================================
UNIT 3: SL 3: ICX7650-2X100G 2-port 200G Module
==========================================================================
UNIT 4: SL 1: ICX7650-48ZP-L3-BASE POE 48-port Management Module
Serial #:EZC3322N01R
License: ICX7650_L3_SOFT_PACKAGE (LID: gbeIIHHpFGt)
==========================================================================
UNIT 4: SL 2: ICX7600-1X100G 1-port 100G Module
Serial #:EZH3335N007
==========================================================================
UNIT 4: SL 3: ICX7650-2X100G 2-port 200G Module
==========================================================================
2000 MHz ARM processor ARMv8 88 MHz bus
8192 KB boot flash memory
2048 MB code flash memory
3910 MB DRAM
Monitor Option is on
STACKID 2 system uptime is 23 day(s) 5 hour(s) 11 minute(s) 25 second(s)
STACKID 1 system uptime is 23 day(s) 5 hour(s) 11 minute(s) 24 second(s)
STACKID 3 system uptime is 23 day(s) 5 hour(s) 11 minute(s) 18 second(s)
STACKID 4 system uptime is 23 day(s) 5 hour(s) 11 minute(s) 26 second(s)
```

```
The system started at 09:14:44 GMT+00 Fri Feb 18 2000
The system : started=warm start reloaded=by "reload"
My stack unit ID = 2, bootup role = active
======================================================
======= WARNING: Boot-monitor version mismatch!!! =======
======= Please use "show boot-monitor " for details =======
======================================================
```

# History

| Release version | Command history |
|---|---|
| 08.0.30j | The output of the **show version** command is updated when a module is removed form the device. |
| 08.0.60 | The command has been modified to show both original factory-installed license and current license. |
| 08.0.61 | The command has been modified to add license information for individual stack units. |
| 08.0.80 | The output prompts you with a warning message if there is a mismatch with the recommended u-boot version. |

# show vlan

Displays the VLAN information.

## Syntax

**show vlan** [ *vlan-id* [ *num* ] | **brief** { **ethernet** *unit/slot/port* | **lag** *lag-id* } | **ethernet** *unit/slot/port* | **lag** *lag-id* ]

## Parameters

*vlan-id*

Specifies the VLAN ID.

*num*

Specifies the number of Layer 3 VLAN entries to skip before the display begins.

**brief**

Displays the VLAN information summary.

**ethernet** *unit/slot/port*

Specifies the Ethernet port for which you want to view VLAN details.

**lag** *lag-id*

Specifies the LAG virtual interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

## Command Output

The **show vlan brief** command displays the following information:

| Output field | Description |
|---|---|
| System-max vlan Params | The system maximum VLAN values (maximum, default, and current). |
| Default vlan Id | The default VLAN ID number. |
| Total Number of Vlan Configured | The total number of VLANs configured on the device. |
| VLANs Configured | The VLAN ID numbers of the VLANs configured on the device. |
| Untagged VLAN | The number of untagged VLANs. |

# Examples

The following example shows the VLAN information for a specific VLAN.

```
device> show vlan 100

Total PORT-VLAN entries: 2
Maximum PORT-VLAN entries: 64
Legend: [Stk=Stack-Id, S=Slot]
PORT-VLAN 100, Name [None], Priority level0, Spanning tree On
 Untagged Ports: None
   Tagged Ports: (U1/M1)   1   2   3   4   5   6   7   8   9  10  11  12
   Tagged Ports: (U1/M1)  13  14  15  16
   Tagged Ports: (LAG)    1   5   15  256
   Uplink Ports: None
 DualMode Ports: None
 Mac-Vlan Ports: None
     Monitoring: Disabled
```

The following example shows the output of the **show vlan** command.

```
device> show vlan

Total PORT-VLAN entries: 4
Maximum PORT-VLAN entries: 4060
Legend: [Stk=Stack-Unit, S=Slot]
PORT-VLAN 1, Name DEFAULT-VLAN, Priority level0, Spanning tree On
Untagged Ports: (Stk0/S1) 3  4  5  6  7  8  9  10 11 12 13 14
Untagged Ports: (Stk0/S1) 15 16 17 18 19 20 21 22 23 24 25 26
Untagged Ports: (Stk0/S1) 27 28 29 30 31 32 33 34 35 36 37 38
Untagged Ports: (Stk0/S1) 39 40 41 42 43 44 45 46 47 48
Untagged Ports: (Stk0/S2) 1  2
Tagged Ports: None
Uplink Ports: None
DualMode Ports: None
Mac-Vlan Ports: None
Monitoring: Disabled
PORT-VLAN 10, Name [None], Priority level0, Spanning tree On
Untagged Ports: (Stk0/S1) 1
Tagged Ports: None
Uplink Ports: None
DualMode Ports: None
Mac-Vlan Ports: None
Monitoring: Enabled
PORT-VLAN 20, Name [None], Priority level0, Spanning tree On
Untagged Ports: (Stk0/S1) 2
Tagged Ports: None
Uplink Ports: None
DualMode Ports: None
Mac-Vlan Ports: None
Monitoring: Disabled
```

The following example shows the output of the **show vlan brief** command.

```
device> show vlan brief

System-max vlan Params: Max(4095) Default(64) Current(3210)
Default vlan Id :1
Total Number of Vlan Configured :5
VLANs Configured :1 to 4 10
```

The following example shows the output of the port-based **show vlan brief ethernet** command. The output indicates the membership type of the VLAN.

```
device(config-if-e1000-1/1/5)#show vlan brief e 1/1/5
Port 1/1/5 is a member of 11 VLANs
VLANs 100 to 110
Untagged VLAN      : <empty>
Tagged VLANs    : 100 to 110
```

The following example shows the output of the port-based **show vlan brief ethernet** command for a flexible authentication port.

```
device> show vlan brief ethernet 1/1/2

Port 1/1/2 is a member of 2 VLANs
VLANs 200 3000
MAC    VLANs : 200
Tagged VLANs : 3000
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was modified to display the VLAN membership type in the **show vlan brief ethernet** command output. |
| 08.0.61 | This command was modified to add the *lag-id* option. |
| 08.0.70 | The **show vlan brief ethernet** command was modified to display the VLAN membership type. |

# show vlan-group

Displays the VLAN group configuration information.

## Syntax

**show vlan-group** [ *group-id* ]

## Parameters

*group-id*

Displays the VLAN group configuration information for the specified VLAN group ID.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

If you do not specify a group ID, the configuration information for all the configured VLAN groups is displayed.

## Examples

The following example displays sample output of the **show vlan-group** command.

```
device# show vlan-group
vlan-group 1 vlan 2 to 20
tagged ethe 1/1/1 to 1/1/2
!
vlan-group 2 vlan 21 to 40
tagged ethe 1/1/1 to 1/1/2
!
```

The following example displays sample output of the **show vlan-group** command for a specific group ID.

```
device# show vlan-group 10
vlan-group 10 vlan 11 to 16
!
```

# show voice-vlan

Displays the configuration of a voice VLAN for a particular port or for all ports.

## Syntax

**show voice-vlan** [ **ethernet** *stack-id/slot/port* ]

## Parameters

**ethernet** *stack-id/slot/port*

Displays the voice VLAN configuration for the specified Ethernet interface.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

## Examples

The following is sample output from the **show voice-vlan** command for all ports.

```
device# show voice-vlan

Port ID         Voice-vlan
1/1/2              1001
1/1/8              150
1/1/15             200
```

The following is sample output from the **show voice-vlan** command for a specific port.

```
device# show voice-vlan ethernet 1/1/2

Voice vlan ID for port 1/1/2: 1001
```

# show vrf

Displays IP information for the specified Virtual Routing and Forwarding (VRF).

## Syntax

**show vrf** [ *vrf-name* | **detail** | **resource** [ **detail** ] ]

## Parameters

*vrf-name*

Specifies the VRF for which you want to display the information.

**detail**

Displays detailed VRF instance information. When used along with the **resource** keyword, displays detailed resource information.

**resource**

Displays resources used by all VRFs.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

VRF configuration mode

## Command Output

The **show vrf** command displays the following information.

| Output field | Description |
|---|---|
| VRF *vrf-name* | The name of the VRF. |
| default RD | The default route distinguisher for the VRF. |
| Table ID | The table ID for the VRF. |
| Routes | The total number of IPv4 and IPv6 unicast routes configured on this VRF. |
| Configured as management-vrf | Indicates that the specified VRF is configured as a management VRF. |
| IP Router-Id | The 32-bit number that uniquely identifies the router. |
| Number of Unicast Routes | The number of unicast routes configured on this VRF. |

# Examples

The following is sample output from the **show vrf** *vrf-name* command.

```
device(config)# show vrf mvrf

VRF mvrf, default RD 1100:1100, Table ID 11
Configured as management-vrf
IP Router-Id: 10.0.0.1
Interfaces:
ve3300 ve3400
Address Family IPv4
Max Routes: 641
Number of Unicast Routes: 2
Address Family IPv6
Max Routes: 64
Number of Unicast Routes: 2
```

# show vsrp

Displays the VSRP information.

## Syntax

**show vsrp** [ **aware** ] [ **vlan** *vlan-id* [ *vrid-num* ] | **vrid** *vrid-num* ]

**show vsrp** [ **brief** ]

## Parameters

**aware**

Displays information about VSRP-aware devices.

**vlan** *vlan-id*

Displays VSRP information for the VLAN ID.

**vrid** *vrid-num*

Displays information for the ports with VSRP enabled.

**brief**

Displays the VSRP information summary.

## Modes

User EXEC mode

## Command Output

The **show vsrp** command displays the following information:

| Output field | Description |
|---|---|
| Total number of VSRP routers defined | The total number of VRIDs configured on this device. |
| VLAN | The VLAN on which VSRP is configured. |
| auth-type | The authentication type in effect on the ports in the VSRP VLAN. |
| VRID | The VRID for which the VSRP information is displayed. |
| state | The device VSRP state for the VRID. The state can be one of the following:<br>• initialize: The VRID is not enabled (activated). If the state remains "initialize" after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other.<br>• standby: This device is a backup for the VRID.<br>• master: This device is the master for the VRID. |
| Administrative-status | The administrative status of the VRID. The administrative status can be one of the following:<br>• disabled: The VRID is configured on the interface but VSRP or VRRP-E has not been activated on the interface.<br>• enabled: VSRP has been activated on the interface. |

| Output field | Description |
|---|---|
| Advertise-backup | Whether the device is enabled to send VSRP Hello messages when it is a backup. This field can have one of the following values:<br>• disabled: The device does not send Hello messages when it is a backup.<br>• enabled: The device sends Hello messages when it is a backup. |
| Preempt-mode | Whether the device can be preempted by a device with a higher VSRP priority after this device becomes the master. This field can have one of the following values:<br>• disabled: The device cannot be preempted.<br>• enabled: The device can be preempted. |
| save-current | The source of VSRP timer values preferred when you save the configuration. This field can have one of the following values:<br>• false: The timer values configured on this device are saved.<br>• true: The timer values most recently received from the master are saved instead of the locally configured values. |
| Configured | Indicates the parameter value configured on this device. |
| Current | Indicates the parameter value received from the master. |
| Unit | Indicates the formula used tor calculating the VSRP priority and the timer scales in effect for the VSRP timers. A timer true value is the value listed in the Configured or Current field divided by the scale value. |
| priority | The device preferability for becoming the master for the VRID. During negotiation, the backup with the highest priority becomes the master. If two or more backups are tied with the highest priority, the backup interface with the highest IP address becomes the master for the VRID. |
| hello-interval | The number of seconds between Hello messages from the master to the backups for a given VRID. |
| dead-interval | The configured value for the dead interval. The dead interval is the number of seconds a backup waits for a Hello message from the master for the VRID before determining that the master is no longer active. If the master does not send a Hello message before the dead interval expires, the backups negotiate (compare priorities) to select a new master for the VRID. If the value is 0, then you have not configured this parameter. |
| hold-interval | The number of seconds a backup that intends to become the master will wait before actually beginning to forward Layer 2 traffic for the VRID. If the backup receives a Hello message with a higher priority than its own before the hold-down interval expires, the backup remains in the backup state and does not become the new master. |
| initial-ttl | The number of hops a Hello message can traverse after leaving the device before the Hello message is dropped. A metro ring counts as one hop, regardless of the number of nodes in the ring. |
| next hello sent in | The amount of time until the master dead interval expires. If the backup does not receive a Hello message from the master by the time the interval expires, either the IP address listed for the master will change to the IP address of the new master, or this Layer 3 switch itself will become the master. This field applies only when this device is a backup. |
| Member ports | The ports in the VRID. |
| Operational ports | The member ports that are currently up. |
| Forwarding ports | The member ports that are currently in the forwarding state. Ports that are forwarding on the master are listed. Ports on the Standby, which are in the blocking state, are not listed. |

The **show vsrp aware** command displays the following information:

| Output field | Description |
|---|---|
| Last Port | The most recent active port connection to the VRID. This is the port connected to the current master. If a failover occurs, the VSRP-aware device changes the port to the port connected to the |

| Output field | Description |
|---|---|
| | new master. The VSRP-aware device uses this port to send and receive data through the backed-up node. |

# Examples

The following example shows the output of the **show vsrp aware** command.

```
device# show vsrp aware
Aware port listing
VLAN ID    VRID    Last Port
100        1       1/3/2
200        2       1/4/1
```

The following example shows the output of the **show vsrp vlan** *vlan-id* **vrid** *vrid-num* command.

```
device# show vsrp vlan 100 vrid 100
VLAN 100
auth-type no authentication
VRID 100
========
State     Administrative-status    Advertise-backup    Preempt-mode    save-current
master    enabled                  disabled            true            false
Parameter          Configured       Current            Unit/Formula
priority           100               50                (100-0)*(2.0/4.0)
hello-interval     1                 1                 sec/1
dead-interval      3                 3                 sec/1
hold-interval      3                 3                 sec/1
initial-ttl        2                 2                 hops
next hello sent in 00:00:00.3
Member ports:      ethe 1/2/5 to 1/2/8
Operational ports: ethe 1/2/5 ethe 1/2/8
Forwarding ports:  ethe 1/2/5 ethe 1/2/8
Restart ports:     1/2/5(1) 1/2/6(1) 1/2/7(1) 1/2/8(1)
```

# show vxlan tunnel

Displays overlay gateway tunnel information.

## Syntax

**show vxlan tunnel** [**brief** | *tunnel_id* | **statistics** ]

## Parameters

brief
: Displays brief information for all VXLAN tunnels.

*tunnel_id*
: Displays detailed information for specified VXLAN tunnel.

**statistics**
: Displays counters for all VXLAN tunnels.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

VLAN configuration mode

## Command Output

The **show vxlan tunnel** command displays the following information:

| Output field | Description |
|---|---|
| | |
| | |
| | |

## Examples

The following example shows a sample output of the **show vxlan tunnel** command.

```
device#
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | The command was introduced. |

# show webauth

Displays Web Authentication configuration details.

## Syntax

**show webauth** [ **allowed-list** | **authenticating-list** | **blocked-list** | **vlan** *vlan-id* [ **passcode** | **webpage** ] ]

## Parameters

**allowed-list**
Displays a list of hosts that are currently authenticated.

**authenticating-list**
Displays a list of hosts that are trying to authenticate.

**blocked-list**
Displays a list of hosts that are currently blocked from any Web Authentication attempt.

**vlan** *vlan-id*
Displays Web Authentication details on a specific VLAN.

**passcode**
Displays current dynamic passcode details.

**webpage**
Displays what text has been configured for Web Authentication pages.

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

Interface configuration mode

Web Authentication configuration mode

## Usage Guidelines

The **show webauth** command by itself displays information for all VLANs on which Web Authentication is enabled.

## Command Output

The **show webauth** command displays the following information:

| Output field | Description |
|---|---|
| WEB AUTHENTICATION (VLAN #) | Identifies the VLAN on which Web Authentication is enabled. |
| attempt-max-num | The maximum number of Web Authentication attempts during a cycle. |
| host-max-num | The maximum number of users that can be authenticated at one time. |

| Output field | Description |
|---|---|
| block duration | The number of seconds a user who failed Web Authentication must wait before attempting to be authenticated. |
| cycle-time | The number of seconds in one Web Authentication cycle. |
| port-down-authenticated-mac-cleanup | Indicates if this option is enabled or disabled. If enabled, all authenticated users are deauthenticated if all the ports in the VLAN go down. |
| reauth-time | The number of seconds an authenticated user remains authenticated. Once this timer expires, the user must reauthenticate. |
| authenticated-mac-age-time | If a user is inactive, this time shows how many seconds a user has before the user-associated MAC address is aged out. The user will be forced to reauthenticate. |
| dns-filter | Shows the definition of any DNS filter that has been set. |
| authentication mode | The authentication mode: username and password (default), passcode, captive-portal, or none. Also displays configuration details for the authentication mode. |
| RADIUS accounting | Whether RADIUS accounting is enabled or disabled. |
| Trusted port list | The statically-configured trusted ports of the Web Authentication VLAN. |
| Secure login (HTTPS) | Whether HTTPS is enabled or disabled. |
| Web Page Customizations | The current configuration for the text that appears on the Web Authentication pages. Either "Custom Text" or "Default Text" displays for each page type:<br>• "Custom Text" means the message for the page has been customized. The custom text is also displayed.<br>• "Default Text" means the default message that ships with the device is used.<br>The actual text on the Web Authentication pages can be displayed using the **show webauth vlan** *vlan-id* **webpage** command. |
| Host statistics | The authentication status and the number of hosts in each state. |

The **show webauth allowed-list** command displays the following information:

| Output field | Description |
|---|---|
| VLAN #: Web Authentication | The ID of the VLAN on which Web Authentication is enabled. |
| Web Authenticated List MAC Address | The MAC addresses that have been authenticated. |
| AuthMode | The client is authenticated using internal server or external server. |
| User Name | The authenticated username. |
| Configuration Static/Dynamic | If the MAC address was dynamically (passed Web Authentication) or statically (added to the authenticated list using the **add mac** command) authenticated. |
| Authenticated Duration HH:MM:SS | The remainder of time the MAC address will remain authenticated. |
| Dynamic ACL | The dynamically assigned ACL. |

The **show webauth authenticating-list** command displays the following information:

| Output field | Description |
|---|---|
| VLAN #: Web Authentication | The ID of the VLAN on which Web Authentication is enabled. |
| MAC Address | The MAC addresses that are trying to be authenticated. |
| AuthMode | The client is authenticated using internal server or external server. |
| User Name | The User Name associated with the MAC address. |
| # of Failed Attempts | Number of authentication attempts that have failed. |

| Output field | Description |
|---|---|
| Cycle Time Remaining | The remaining time the user has to be authenticated before the current authentication cycle expires. Once it expires, the user must enter a valid URL again to display the Web Authentication Welcome page. |

The **show webauth blocked-list** command displays the following information:

| Output field | Description |
|---|---|
| VLAN #: Web Authentication | The ID of the VLAN on which Web Authentication is enabled. |
| Web Block List MAC Address | The MAC addresses that have been blocked from Web Authentication. |
| AuthMode | The client is authenticated using internal server or external server. |
| User Name | The username associated with the MAC address. |
| Configuration Static/Dynamic | If the MAC address was dynamically or statically blocked. The **block mac** command statically blocks MAC addresses. |
| Block Duration Remaining | The remaining time the MAC address has before the user with that MAC address can attempt Web Authentication. |

# Examples

The following example displays sample output of the **show webauth** command.

```
device# show webauth
================================================================================
WEB AUTHENTICATION (VLAN 25): Enable
attempt-max-num: 5 (Default)
host-max-num: 0 (Default)
block duration: 90 (Default)
cycle-time: 600 (Default)
port-down-authenticated-mac-cleanup: Enable (Default)
reauth-time: 28800 (Default)
authenticated-mac-age-time: 3600 (Default)
dns-filter: Disable (Default)
authentication mode: username and password (Default)
  authentication methods: radius
   Local user database name: <none>
Radius accounting: Enable (Default)
Trusted port list: None
Secure Login (HTTPS): Enable (Default)
Web Page Customizations:
 Top (Header): Default Text
 Bottom (Footer): Custom Text
  "SNL Copyright 2009"
 Title: Default Text
 Login Button: Custom Text
  "Sign On"
 Web Page Logo: blogo.gif
  align: left (Default)
 Web Page Terms and Conditions: policy1.txt
Host statistics:
 Number of hosts dynamically authenticated: 0
 Number of hosts statically authenticated: 2
 Number of hosts dynamically blocked: 0
 Number of hosts statically blocked: 0
 Number of hosts authenticating: 1
```

The following example displays sample output of the **show webauth allowed-list** command.

```
device# show webauth allowed-list
================================================================================
VLAN 3: Web Authentication, Mode: I = Internal E = External
--------------------------------------------------------------------------------
Web Authenticated List                Configuration   Authenticated Duration  Dynamic
MAC Address       User Name   mode    Static/Dynamic  HH:MM:SS                ACL
--------------------------------------------------------------------------------
000c.2973.a42b    ruckus      E       D               1 day, 11:33:16         acl1
1222.0a15.f045    super       E       D               1 day, 11:32:51         acl1
1222.0a15.f044    foundry     E       D               1 day, 11:32:48         acl1
1222.0a15.f043    ruckus      E       D               1 day, 11:32:47         acl1
1222.0a15.f042    spirent     E       D               1 day, 11:32:4          acl1
```

The following example displays sample output of the **show webauth authenticating-list** command.

```
device# show webauth authenticating-list
================================================================================
VLAN 3: Web Authentication, AuthMode: I=Internal E=External
--------------------------------------------------------------------------------
Web Authenticating List             # of Failed  Cycle Time Remaining
MAC Address        User Name  mode  Attempts     HH:MM:SS
--------------------------------------------------------------------------------
000c.2973.a42b     N/A        E     0            00:01:36
```

The following example displays sample output of the **show webauth blocked-list** command.

```
device# show webauth blocked-list
================================================================================
VLAN 3: Web Authentication, AuthMode: I=Internal E=External
--------------------------------------------------------------------------------
Block List                        Configuration mode  Block Duration Remaining
MAC Address     User Name  mode   Static/Dynamic
--------------------------------------------------------------------------------
000c.2973.a42b  User1      E      D                   00:00:04
```

The following example displays sample output of the **show webauth vlan** *vlan-id* **passcode** command.

```
device# show webauth vlan 25 passcode
Current Passcode : 1389
This passcode is valid for 35089 seconds
```

The following is a sample output of the **show webauth vlan** *vlan-id* **webpage** command.

```
device# show webauth vlan 25 webpage
==================================================
Web Page Customizations (VLAN 25):
  Top (Header): Default Text
    "<h3>Welcome to Ruckus Networks Web Authentication Homepage</h3>"
  Bottom (Footer): Custom Text
    "Copyright 2009 SNL"
  Title: Default Text
    "Web Authentication"
  Login Button: Custom Text
    "Sign On"
  Web Page Logo: blogo.gif
    align: left (Default)
  Web Page Terms and Conditions: policy1.txt
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | The output was modified to include "mode" and "Dynamic ACL" fields. |

# show who

Displays details of the SSH and Telnet connections.

## Syntax

**show who**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Examples

The following example displays output of the **show who** command.

```
device(config)# show who
Console connections:
        established, privilege super-user, in config mode
        you are connecting to this session
        12 seconds in idle
Telnet server status: Enabled
Telnet connections (inbound):
 1      closed
 2      closed
 3      closed
 4      closed
 5      closed
Telnet connections (outbound):
 6      closed
 7      closed
 8      closed
 9      closed
 10     closed
SSH server status: Disabled
SSH copy-received-cos status: Disabled
SSH connections:
SSH connections (inbound):
 1      closed
 2      closed
 3      closed
 4      closed
 5      closed
SSH connection (outbound):
 6      closed
 7      closed
 8      closed
 9      closed
 10     closed
```

# Commands Sn - Z

## slow-start

Configures a slow-start timer interval to extend the time interval beyond the dead-interval time before a Virtual Router Redundancy Protocol Extended (VRRP-E) master device assumes the role of master device after being offline. When the original master device went offline, a backup VRRP-E device with a lower priority became the master device.

### Syntax

**slow-start** *seconds*

**no slow-start** *seconds*

### Command Default

If a slow-start timer is not configured, the master device assumes control from a backup device immediately after the dead interval.

### Parameters

*seconds*

Sets the number of seconds for the slow-start timer. Range from 1 through 57600.

### Modes

VRRP-E router configuration mode

### Usage Guidelines

When the VRRP-E slow-start timer is enabled, if the master VRRP-E device goes down, the backup device with the highest priority takes over after the expiration of the dead interval. If the original master device subsequently comes back up again, the amount of time specified by the VRRP-E slow-start timer elapses before the original master device takes over from the backup device (which became the master device when the original master device went offline).

This command is supported only for VRRP-E.

The **no** form removes the slow-start configuration.

### Examples

The following example sets the slow-start timer interval to 40 seconds.

```
device# configure terminal
device(config)# router vrrp-extended
device(config-vrrpe-router)# slow-start 40
```

# snmp-client

Restricts SNMP access to a host with the specified IPv4 or IPV6 address.

## Syntax

**snmp-client** {*ip-address* | **ipv6** *ipv6-address* }

**no snmp-client** {*ip-address* | **ipv6** *ipv6-address* }

## Command Default

SNMP access is not restricted.

## Parameters

*ip-address*
The IPv4 address of the host to which the SNMP access is restricted.

**ipv6** *ipv6-address*
Specifies the IPv6 address of the host to which the SNMP access is restricted.

## Modes

Global configuration mode

## Usage Guidelines

You can specify only one IP address with one command. However, you can enter the command ten times to specify up to ten IP addresses.

The **no** form of the command removes the SNMP access restriction.

## Examples

The following example shows how to allow SNMP access only to the host with IP address 192.168.10.1.

```
device(config)# snmp-client 192.168.10.1
```

# snmp-server community

Configures the SNMP community string and access privileges.

## Syntax

**snmp-server community** *community-string* { **ro** | **rw** } [ *acl-name* | *acl-num* | **ipv6** *ipv6-acl-name* | **view** [ *mib-view* ] ]

**no snmp-server community** *community-string* { **ro** | **rw** } [ *acl-name* | *acl-num* | **ipv6** *ipv6-acl-name* | **view** [ *mib-view* ] ]

## Command Default

The SNMP community string is not configured.

## Parameters

*community-string*
: Configures the SNMP community string that you must enter to gain SNMP access. The string is an ASCII string and can have up to 32 characters.

**ro**
: Configures the community string to have read-only ("get") access.

**rw**
: Configures the community string to have read-write ("set") access.

*acl-name*
: Filters incoming packets using a named standard access control list (ACL).

*acl-num*
: Filters incoming packets using a numbered ACL.

**ipv6** *ipv6-acl-name*
: Filters incoming packets using a named IPv6 ACL.

**view** *mib-view*
: Associates a view to the members of the community string. Enter up to 32 alphanumeric characters.

## Modes

Global configuration mode

## Usage Guidelines

The **view** *mib-view* parameter allows you to associate a view to the members of this community string. If no view is specified, access to the full MIB is granted. The view that you want must exist before you can associate it to a community string.

You can set just one access type, either read-only (ro) or read/write (rw) for a single SNMP community instead of setting both access types. The read/write access supersedes read-only configuration and if read/write is configured for a specified community after read only, the running configuration file only saves the rw configuration line.

If you issue the **no snmp-server community public ro** command and then enter the **write memory** command to save the configuration, the read-only "public" community string is removed and will have no SNMP access. If for some reason the device is brought down and then brought up, the **no snmp-server community public ro** command is restored in the system and the read-only "public" community string has no SNMP access.

The **no** form of the command removes an SNMP community string.

# Examples

The following example configures an SNMP community string with read-only access.

```
device# configure terminal
device(config)# snmp-server community private ro
```

The following example configures an ACL to filter SNMP packets.

```
device# configure terminal
device(config)# access-list 25 deny host 10.157.22.98 log
device(config)# access-list 25 deny 10.157.23.0 0.0.0.255 log
device(config)# access-list 25 deny 10.157.24.0 0.0.0.255 log
device(config)# access-list 25 permit any
device(config)# access-list 30 deny 10.157.25.0 0.0.0.255 log
device(config)# access-list 30 deny 10.157.26.0/24 log
device(config)# access-list 30 permit any
device(config)# snmp-server community public ro 25
device(config)# snmp-server community private rw 30
device(config)# write memory
```

The following example associates a view to the members of a community string.

```
device# configure terminal
device(config)# snmp-server community private rw view view1
```

# snmp-server contact

Configures the identification of the contact person for the managed node.

## Syntax

**snmp-server contact** *name*

**no snmp-server contact** *name*

## Command Default

Contact information is not configured.

## Parameters

*name*

The contact name. The name can be up to 255 alphanumeric characters. Spaces are allowed.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the contact information.

## Examples

The following example configures the identification of the contact person for the device.

```
device(config)# snmp-server contact Sales
```

# snmp-server disable

Disables SNMP MIB support.

## Syntax

**snmp-server disable mib** *table*

**no snmp-server disable mib** *table*

## Command Default

SNMP MIB support is enabled.

## Parameters

**mib** *table*

Disables MIB support for a given table. Support for the following tables can be disabled:

**dot1d-tp-fdb**

Disables SNMP support for dot1dTpFdbTable.

**dot1q-fdb**

Disables SNMP support for dot1qFdbTable.

**dot1q-tp-fdb**

Disables SNMP support for dot1qTpFdbTable.

**enet-pw**

Disables SNMP support for pwEnetTable.

**pw**

Disables SNMP support for pwTable.

**vll-ep**

Disables SNMP support for fdryVllEndPointTable.

**vpls-ep-vlan-ext**

Disables SNMP support for brcdVplsEndptVlanExtStatsTable.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command enables SNMP MIB support.

# Examples

The following example disables dot1d-tp-fdb MIB support.

```
decice(config)# snmp-server disable mib dot1d-tp-fdb
```

# snmp-server enable

Configures SNMP access only to specific clients.

## Syntax

**snmp-server enable ethernet** *stack/slot/port* [ **to** *stack/slot/port* | [ **ethernet** *stack/slot/port* **to** *stack/slot/port* | **ethernet** *stack/slot/port* ] ... ]

**no snmp-server enable ethernet** *stack/slot/port* [ **to** *stack/slot/port* | [ **ethernet** *stack/slot/port* **to** *stack/slot/port* | **ethernet** *stack/slot/port* ] ... ]

**snmp-server enable vlan** *vlan-id*

**no snmp-server enable vlan** *vlan-id*

## Command Default

SNMP access is not restricted.

## Parameters

**ethernet** *stack/slot/port*
Specifies the Ethernet interface on which web management should be enabled.

**to** *stack/slot/port*
Specifies the range of Ethernet interfaces.

**vlan** *vlan-id*
Specifies that web management should be enabled on the clients of the specified VLAN.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the SNMP access restriction.

## Examples

The following example configures the SNMP access only to a client in VLAN 40.

```
device(config)# snmp-server enable vlan 40
```

The following example configures SNMP access to a range of Ethernet interfaces.

```
device(config)# snmp-server enable ethernet 1/1/1 to ethernet 1/1/5
```

# snmp-server enable mib

Enables MIB support for SNMP server.

## Syntax

**snmp-server enable mib** *mib-name*

**no snmp-server enable mib** *mib-name*

## Command Default

MIB support is enabled by default.

## Parameters

*mib-name*

Enables support for one of the following MIBs:

**np-qos-stat**

Enables SNMP support for brcdNPQosStatTable.

**tm-dest-qstat**

Enables SNMP support for brcdTMDestUcastQStatTable.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command disables the SNMP MIB support.

## Examples

The following example enables the brcdTMDestUcastQStatTable MIB support.

```
device(config)# snmp-server enable mib tm-dest-qstat
```

# snmp-server enable traps

Enables SNMP traps for various events.

## Syntax

**snmp-server enable traps** *event*

**no snmp-server enable traps** *event*

## Command Default

Traps are enabled by default.

## Parameters

*event*

The event for which the traps should be enabled. Enables the traps for one of the following events:

**authentication**

Generates the trap when the authentication occurs.

**cold-start**

Generates the trap after a cold start.

**fan-failure**

Generates the trap when there is a fan failure and when the issue is resolved.

**fan-speed-change**

Generates the trap when there is a change in fan speed.

**ikev2**

Generates the trap for Internet Key Exchange Protocol, v2 (IKEv2) events.

**ipsec**

Generates the trap for Internet Protocol Security (IPsec) events.

**link-down**

Generates the trap when the link is down.

**link-oam**

Generates the trap for link OAM.

**link-up**

Generates the trap when the link is up.

**mac-authentication**

Generates the trap when a MAC address is added or deleted.

**mac-notification**

Generates the trap after a MAC authentication.

**metro-ring**

Generates the trap when there is a change in the Metro Ring configuration.

**module-inserted**
Generates the trap when a module is inserted.

**module-removed**
Generates the trap when a module is removed.

**new-root**
Generates a control STP trap for newRoot events, as defined in RFC 1493.

**nlp-phy-40g**
Generates the trap during PHY calibration on the 40-Gbps and 4x10-Gbps stack ports.

**power-supply-failure**
Generates the trap when there is a power supply failure and when the issue is resolved.

**redundant-module**
Generates thea control enterprise trap snTrapMgmtModuleRedunStateChange for redundant module events..

**syslog**
Generates syslogMsgNotification traps.

**temperature**
Generates the trap when there is a temperature change.

**topology-change**
Generates the control STP trap topologyChange defined in RFC 1493 for topology changes.

**udld**
Generates the control enterprise traps for Unidirectional Link Detection (UDLD) events.

**vsrp**
Generates the control enterprise Virtual Switch Redundancy Protocol (VSRP) trap snTrapVsrpIfStateChange.

# Modes

Global configuration mode

# Usage Guidelines

The **no** form of the command disables the traps.

The **ipsec** and **ikev2** options are only supported on the Ruckus ICX 7450, with an FPGA-based add-on crypto card.

# Examples

The following example enables SNMP traps on the device for MAC notification globally.

```
device(config)# snmp-server enable traps mac-notification
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | The **mac-notification** keyword was added. |
| 8.0.50 | The **ipsec** and **ikev2** keywords were added. |

# snmp-server enable traps holddown-time

Configures the wait time before starting to send SNMP traps.

## Syntax

**snmp-server enable traps holddown-time** *time*

**no snmp-server enable traps holddown-time** *time*

## Command Default

The default hold-down time is 60 seconds.

## Parameters

*time*

The time in seconds. The valid range is from 1 through 600 seconds. The default is 60 seconds.

## Modes

Global configuration mode

## Usage Guidelines

When a device starts up, the software waits for Layer 2 convergence (STP) and Layer 3 convergence (OSPF) before beginning to send SNMP traps to external SNMP servers. Until convergence occurs, the device may not be able to reach the servers, in which case the messages are lost.

By default, a device uses a one-minute hold-down time to wait for the convergence to occur before starting to send SNMP traps. After the hold-down time expires, the device sends the traps, including traps such as "cold start" or "warm start" that occur before the hold-down time expires.

When you have a stack of eight or more units, you must increase the trap hold-down time from the default (60 seconds) to five minutes (300 seconds). This will prevent the loss of initial boot traps.

The **no** form of the command changes the hold-down time to the default value.

## Examples

The following example changes the hold-down time for SNMP traps to 30 seconds.

```
device(config)# snmp-server enable traps holddown-time 30
```

# snmp-server enable traps mac-notification

Enables the MAC-notification trap whenever a MAC address event is generated on a device or an interface.

## Syntax

**snmp-server enable traps mac-notification**

**no snmp-server enable traps mac-notification**

## Command Default

MAC-notification traps are disabled on the device.

## Modes

Global configuration

Interface configuration

## Usage Guidelines

The **no** form of this command disables SNMP traps for MAC-notification events. The SNMP MAC-notification trap functionality allows an SNMPv3 trap to be sent to the SNMP manager when MAC addresses are added or deleted in the device.

## Examples

The following example enables SNMP traps on the device for MAC-notification globally:

```
device(config)# snmp-server enable traps mac-notification
```

The following example disables SNMP traps on the device for MAC-notification globally:

```
device(config)# no snmp-server enable traps mac-notification
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# snmp-server engineid local

Modifies the default SNMPv3 engine ID.

## Syntax

**snmp-server engineid local** *engineid-string*

**no snmp-server engineid local** *engineid-string*

## Command Default

A default engine ID is generated during system startup.

## Parameters

*engineid-string*
Specifies the engine ID as a hexidecimal character string with an even number of characters.

## Modes

Global configuration mode

## Usage Guidelines

The default engine ID guarantees the uniqueness of the engine ID for SNMP version 3. A default engine ID is generated during system startup. To determine the default engine ID of the device, enter the **show snmp engineid** command. Use the **snmp-server engineid local** command to change the default engine ID.

Each user localized key depends on the SNMP server engine ID, so all users must be reconfigured whenever the SNMP server engine ID changes.

**NOTE**
Because the current implementation of SNMP version 3 does not support Notification, remote engine IDs cannot be configured at this time.

The *engineid-string* variable consists of 11 octets, entered as hexadecimal values. There are two hexadecimal characters in each octet. There must be an even number of hexadecimal characters in an engine ID.

The default engine ID has a maximum of 11 octets:

- Octets 1 through 4 represent the agent's SNMP management private enterprise number as assigned by the Internet Assigned Numbers Authority (IANA). The most significant bit of Octet 1 is "1". For example, "000007c7" is the ID for Ruckus in hexadecimal. With Octet 1 always equal to "1", the first four octets in the default engine ID is always "800007c7" (which is 1991 in decimal).
- Octet 5 is always 03 in hexadecimal and indicates that the next set of values represents a MAC address.
- Octets 6 through 11 form the MAC address of the lowest port in the management module.

The engine ID must be a unique number among the various SNMP engines in the management domain.

The **no** form of the command sets the engine ID to the default.

# Examples

The following example shows how to change the default engine ID.

```
device(config)# snmp-server engineid local 800007c70300e05290ab60
```

# snmp-server group

Creates user-defined groups for SNMPv1/v2c/v3 and configures read, write, and notify permissions to access the MIB view.

## Syntax

**snmp-server group** *groupname* { **v1** | **v2c** } [ **access** { *standard-ACL-id* | **ipv6** *ipv6-ACL-name* } ] [ **notify** *viewname* ] [ **read** *viewname* ] [ **write** *viewname* ]

**no snmp-server group** *groupname* { **v1** | **v2c** } [ **access** { *standard-ACL-id* | **ipv6** *ipv6-ACL-name* } ] [ **notify** *viewname* ] [ **read** *viewname* ] [ **write** *viewname* ]

**snmp-server group** *groupname* **v3** { **auth** | **noauth** | **priv** } [ **access** { *standard-ACL-id* | **ipv6** *ipv6-ACL-name* } ] [ **notify** *viewname* ] [ **read** *viewname* ] [ **write** *viewname* ]

**no snmp-server group** *groupname* **v3** { **auth** | **noauth** | **priv** } [ **access** { *standard-ACL-id* | **ipv6** *ipv6-ACL-name* } ] [ **notify** *viewname* ] [ **read** *viewname* ] [ **write** *viewname* ]

## Command Default

Six default groups are supported to associate the default SNMPv3 user groups and the default SNMPv1/v2c community groups with the view configuration.

**NOTE**
This command is not used for SNMP version 1 and SNMP version 2. In these versions, groups and group views are created internally using community strings. When a community string is created, two groups are created, based on the community string name. One group is for SNMP version 1 packets, while the other is for SNMP version 2 packets.

## Parameters

*groupname*
Specifies the name of the SNMP group to be created.

**v1**
Specifies SNMP version 1.

**v2c**
Specifies SNMP version 2.

**v3**
Specifies SNMP version 3.

**auth**
Specifies that only authenticated packets with no privacy are allowed to access the specified view. This parameter is available only for SNMPv3 user groups.

**noauth**
Specifies that no authentication and no privacy are required to access the specified view. This parameter is available only for SNMPv3 user groups.

**priv**
Specifies that authentication and privacy are required from the users to access the view. This parameter is available only for SNMPv3 user groups.

**access**
Specifies an access list associated with the SNMP group.

*standard-ACL-id*
Specifies the standard IP access list and allows the incoming SNMP packets to be filtered based on the standard ACL attached to the group.

**ipv6**
Specifies the IPv6 ACL for the SNMP group.

*ipv6-ACL-name*
Specifies the IPv6 access list and allows incoming SNMP packets to be filtered based on the IPv6 ACL attached to the group.

**notify** *viewname*
Specifies the name of the view that enables you to provide access to the MIB for trap or inform. This allows the administrators to restrict the scope of varbind objects that will be part of the notification. All of the varbinds need to be in the included view for the notification to be created.

**read** *viewname*
Specifies the name of the view that enables you to provide read access.

**write** *viewname*
Specifies the name of the view that enables you to provide both read and write access.

*viewname*
Specifies the name of the view to which the SNMP group members have access. If no view is specified, then the group has no access to the MIB. The default viewname is "all", which allows access to the entire MIB.

# Modes

Global configuration mode

# Usage Guidelines

Maximum number of SNMP groups supported is 10.

The **no** form of the command removes the configured SNMP server group.

# Examples

The following example creates SNMP server group entries for SNMPv3 user group with auth permission.

```
device(config)# snmp-server group admin v3 auth ipv6 acl_1 read all write all notify all
```

# History

| Release version | Command history |
|---|---|
| 08.0.20a | The **ipv6** *ipv6-ACL-name* keyword-argument pair was introduced. |

# snmp-server host

Configures a trap receiver to ensure that all SNMP traps sent by the Ruckus device go to the same SNMP trap receiver or set of receivers, typically one or more host devices on the network.

## Syntax

**snmp-server host** { *host-ipaddr* | **ipv6** *host-ipv6-addr* } [ **version** { **v1** | **v2c** } [ *community-string* [ **port** *port-num* ] ] ]

**no snmp-server host** { *host-ipaddr* | **ipv6** *host-ipv6-addr* } [ **version** { **v1** | **v2c** } [ *community-string* [ **port** *port-num* ] ] ]

**snmp-server host** { *host-ipaddr* | **ipv6** *host-ipv6-addr* } [ **version v3** { **auth** | **noauth** | **priv** } *name* [ **port** *port-num* ] ]

**no snmp-server host** { *host-ipaddr* | **ipv6** *host-ipv6-addr* } [ **version v3** { **auth** | **noauth** | **priv** } *name* [**port** *port-num* ] ]

## Command Default

The SNMP trap receiver is not configured.

## Parameters

*host-ipaddr*
Specifies the IP address of the trap receiver.

**ipv6** *host-ipv6-addr*
Specifies the IPv6 address of the trap receiver.

**version**
Configures the SNMP version or security model.

**v1**
Specifies SNMP version 1.

**v2c**
Specifies SNMP version 2c.

*community-string*
Specifies an SNMP community string configured on the device.

**v3**
Specifies SNMP version 3.

**auth**
Specifies that only authenticated packets with no privacy are allowed to access the specified view. This parameter is available only for SNMPv3 user groups.

**noauth**
Specifies that no authentication and no privacy are required to access the specified view. This parameter is available only for SNMPv3 user groups.

**priv**
Specifies that authentication and privacy are required from the users to access the view. This parameter is available only for SNMPv3 user groups.

*name*

Specifies the SNMP security name or user.

**port** *port-num*

Configures the UDP port to be used by the trap receiver. The default port number is 162.

# Modes

Global configuration mode

# Usage Guidelines

The device sends all the SNMP traps to the specified hosts and includes the specified community string. Administrators can therefore filter for traps from a device based on IP address or community string. When you add a trap receiver, the software automatically encrypts the community string you associate with the receiver when the string is displayed by the CLI or Web Management interface. The software does not encrypt the string in the SNMP traps sent to the receiver.

The SNMP community string configured can be a read-only string or a read-write string. The string is not used to authenticate access to the trap host but is instead a useful method for filtering traps on the host. For example, if you configure each of your devices that use the trap host to send a different community string, you can easily distinguish among the traps from different devices based on the community strings.

The Multiple SNMP Community Names feature introduced the ability to configure one default community string (where a community string is not mapped to any SNMP context) and one community string per SNMP context for a single trap host. One community name per line is allowed. For protocol-specific MIBS, devices send the trap originating from specific VRF instance and the corresponding community name mapped to the SNMP context associated with that VRF is sent in the trap. When the devices send the trap originating from a default VRF instance, the default community string is sent in the trap. Using the community string in the trap, administrators can easily distinguish among the traps originated from different VRF instances. if you enter the **show running-config** command it displays multiple **snmp-server host** command instances for each host; one community name per line.

Specifying the port allows you to configure several trap receivers in a system. With this parameter, a network management application can coexist in the same system. Devices can be configured to send copies of traps to more than one network management application.

The **no** form of the command removes the configured SNMP server host.

# Examples

The following example configures 10.10.10.1 as the trap receiver.

```
device(config)# snmp-server host 10.10.10.1 version v2c mypublic port 200
```

The following example configures 2002::2:2 as the trap receiver and specifies that only authenticated packets with no privacy are allowed to access the specified view.

```
device(config)# snmp-server host ipv6 2002::2:2 version v3 auth user-private port 110
```

# snmp-server legacy

Configures legacy values for SNMP MIBs.

## Syntax

**snmp-server legacy** { **iftype** | **module-type** }

**no snmp-server legacy** { **iftype** | **module-type** }

## Command Default

SNMP MIBs have the user-configured values.

## Parameters

**iftype**

Configures to the use of legacy Ethernet interface names for ifType.

**module-type**

Configures to the use of legacy enum values for snAgentConfigModuleType.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command changes the settings back to the non-legacy values.

## Examples

The following example configures to the use of legacy Ethernet interface names for ifType.

```
device(config)# snmp-server legacy iftype
```

# snmp-server location

Configures the SNMP server location.

## Syntax

**snmp-server location** *string*

**no snmp-server location** *string*

## Command Default

The SNMP server location is not configured.

## Parameters

*string*

The physical location of the server. The string can be up to 255 alphanumeric characters. Spaces are allowed.

## Modes

Global configuration mode

## Usage Guidelines

You can configure a location for a device and save the information locally in the configuration file for future reference. This information is not required for system operation but is suggested.

The **no** form of the command removes the configured location.

## Examples

The following example configures the physical location of the SNMP server.

```
device(config)# snmp-server location United States
```

# snmp-server max-ifindex-per-module

Configures the maximum number of ifindexes per module.

## Syntax

**snmp-server max-ifindex-per-module** *number*

**no snmp-server max-ifindex-per-module** *number*

## Command Default

The system assigns 64 indexes to each module on the device.

## Parameters

*number*

Specifies the maximum number of ifindexes per module (20, 40 or 64).

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the maximum number of ifindexes per module as 64.

SNMP Management Information Base (MIB) uses Interface Index (ifIndex) to assign a unique value to each port on a module or slot. You can assign 20, 40, and 64 ifindexes per module.

## Examples

The following example configures the number of ifindexes per module to 40.

```
device(config)# snmp-server max-ifindex-per-module 40
```

# snmp-server preserve-statistics

Decouples SNMP statistics from CLI-based statistics.

## Syntax

**snmp-server preserve-statistics**

**no snmp-server preserve-statistics**

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command couples the SNMP statistics from the CLI based statistics.

## Examples

The following example shows how to decouple SNMP statistics from CLI-based statistics.

```
device(config)# snmp-server preserve-statistics
```

# snmp-server pw-check

Controls password check on file operation MIB objects.

## Syntax

**snmp-server pw-check**

**no snmp-server pw-check**

## Command Default

Password check is not configured.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the password check on file operation MIB objects.

Once the password check is enabled, if a third-party SNMP management application does not add a password to the password field when it sends SNMP set requests to a device, by default the device rejects the request.

## Examples

The following example configures password check on file operation MIB objects.

```
device(config)# snmp-server pw-check
```

# snmp-server trap-source

Configures an interface as the source for all traps.

## Syntax

**snmp-server trap-source** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **ve** *number* }

**no snmp-server trap-source** { **ethernet** *stack-id/slot/port* | **loopback** *number* | **ve** *number* }

## Command Default

An SNMP trap generator is not configured.

## Parameters

**ethernet** *stack-id/slot/port*
Specifies an Ethernet interface address to be used as the source for all traps.

**loopback** *number*
Specifies a loopback interface address to be used as the source for all traps.

**ve** *number*
Specifies a Virtual Ethernet interface address to be used as the source for all traps.

## Modes

Global configuration mode

## Usage Guidelines

Regardless of the port that the device uses to send traps to the receiver, the traps always arrive from the same source IP address.

The **no** form of the command removes the configured interface as the SNMP trap generator.

## Examples

The following example configures an Ethernet interface as the SNMP trap generator source.

```
device(config)# snmp-server trap-source ethernet 1/1/1
```

The following example configures a loopback interface as the SNMP trap generator source.

```
device(config)# snmp-server trap-source loopback 10.0.1.1
```

# snmp-server user

Creates or changes the attributes of SNMPv3 users, and allows an SNMPv3 user to be associated with the user-defined group name.

## Syntax

**snmp-server user** *user-name group-name* **v3** [ **access** *acl-num* ] [ **auth** { **md5** | **sha** } *auth-password* [ **priv** { **aes** | **des** } *password-string* ] ]

**no snmp-server user** *user-name group-name* **v3** [ **access** *acl-num* ] [ **auth** { **md5** | **sha** } *auth-password* [ **priv** { **aes** | **des** } *password-string* ] ]

## Command Default

SNMP users are not configured.

## Parameters

*user-name*
Specifies the SNMP username or security name used to access the management module.

*group-name*
Identifies the SNMP group to which this user is associated or mapped.

**v3**
Configures the group using the User Security Model (SNMPv3).

**access**
Specifies the access list associated with the user.

*acl-num*
Standard IP access list number allowing access. The valid values are from 1 through 99.

**auth**
Specifies the type of encryption the user must have to be authenticated.

**md5**
Configures the HMAC MD5 algorithm for authentication.

**sha**
Configures the HMAC SHA algorithm for authentication.

*auth-password*
Specifies the authorization password for the user (8 through 16 characters for MD5; 8 through 20 characters for SHA).

**priv**
Configures the encryption type (DES or AES) used to encrypt the privacy password.

**aes**
Configures CFB128-AES-128 encryption for privacy.

**des**
Configures CBC56-DES encryption for privacy.

*password-string*

Specifies the DES or AES password string for SNMPv3 encryption for the user. The password must have a minimum of 8 characters.

# Modes

Global configuration mode

# Usage Guidelines

The **snmp-server user** command creates an SNMP user, defines the group to which the user will be associated, defines the type of authentication to be used for SNMP access by this user, specifies either the **AES** or **DES** encryption types used to encrypt the privacy password.

All users must be mapped to an SNMP group. Groups are defined using the **snmp-server group** command.

**NOTE**
The SNMP group to which the user account will be mapped should be configured before creating the user accounts; otherwise, the group will be created without any views. Also, ACL groups must be configured before configuring user accounts.

**NOTE**
The ACL specified in a user account overrides the ACL assigned to the group to which the user is mapped. If no ACL is entered for the user account, then the ACL configured for the group will be used to filter packets.

The **priv** parameter specifies the encryption type **(DES or AES)** used to encrypt the privacy password. If the encrypted keyword is used, do the following:

- If **DES** is the privacy protocol to be used, enter **des** followed by a 16-octet DES key in hexadecimal format for the DES-password-key . If you include the encrypted keyword, enter a password string of at least 8 characters.
- If **AES** is the privacy protocol to be used, enter **aes** followed by the AES password key. For a small password key, enter 12 characters. For a big password key, enter 16 characters.

The **no** form of the command removes the SNMP access.

# Examples

The following example configures an SNMP user account.

```
device(config)# snmp-server user user1 admin v3 access 2 auth md5 abc123 priv des xyz123
```

# snmp-server view

Creates an SNMP view.

## Syntax

**snmp-server view** *view-name mib-subtree* { **excluded** | **included** }

**no snmp-server view** *view-name mib-subtree* { **excluded** | **included** }

## Command Default

All MIB objects are automatically excluded from any view unless they are explicitly included.

## Parameters

*view-name*
: Configures the alphanumeric name to identify the view. The names cannot contain spaces.

*mib-subtree*
: Configures the name of the MIB object or family. You can use a wildcard (*) in the numbers to specify a sub-tree family.

**excluded**
: Configures the MIB family identified to be excluded from the view.

**included**
: Configures the MIB family identified to be included in the view.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the configured SNMP view.

SNMP views are named groups of MIB objects that can be associated with user accounts to allow limited access for viewing and modification of SNMP statistics and system configuration. SNMP views can also be used with other commands that take SNMP views as an argument.

MIB objects and MIB sub-trees can be identified by a name or by the numbers called object identifiers (OIDs) that represent the position of the object or sub-tree in the MIB hierarchy.

**NOTE**
All MIB objects are automatically excluded from any view unless they are explicitly included; therefore, when creating views using the **snmp-server view** command, indicate which portion of the MIB you want users to access.

# Examples

The following example assigns the view called "admin" a community string or user group. The "admin" view allows access to the MIB objects that begin with the 1.3.6.1.4.1.1991 object identifier.

```
device(config)# snmp-server view admin 1.3.6.1.4.1.1991 included
```

# source-guard enable

Enables IP Source Guard (IPSG) on a port, per-port per-VLAN, a VLAN or range of VLANs, a virtual interface, or on a range of ports.

## Syntax

**source-guard enable** [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]...

**no source-guard enable** [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]...

## Command Default

IPSG is disabled.

## Parameters

**ethernet** *unit/slot/port*
Specifies the Ethernet interface and the interface ID in the unit/slot/port format.

**to** *stackid/slot/port*
Specifies a range of Ethernet interfaces.

**lag** *lag-id*
Specifies the LAG virtual interface.

**to** *lag-id*
Specifies a range of LAG IDs.

## Modes

Interface configuration mode

VLAN configuration mode

## Usage Guidelines

You can enable IPSG on a range of ports within a given slot only. Enabling IPSG across multiple slots is not supported.

The **no** form of the command disables IPSG on the specified interface.

## Examples

The following example enables IPSG on the Ethernet 1/1/4 interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/4
device(config-if-e10000-1/1/4)# source-guard enable
```

The following example enables IPSG on a specific port in a virtual interface.

```
device(config)# vlan 2
device(config-vlan-2)# tagged ethernet 1/1/1
Added tagged port(s) ethernet 1/1/1 to port-vlan 2
device(config-vlan-2)# router-interface ve 2
device(config-vlan-2)# interface ve 2
device(config-vif-2)# source-guard enable ethernet 1/1/1
```

The following example enables IPSG on a range of ports in the same slot.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# interface ethernet 1/1/21 to 1/1/25
decice(config-mif-1/1/21-1/1/25)# source-guard enable
```

The following error displays if you try to configure ports across multiple slots.

```
device(config)# interface ethernet 1/1/18 to 2/1/18
Error - cannot configure multi-ports on different slot
```

The following example configures IPSG on a range of ports on a VLAN.

```
device# configure terminal
device(config)# vlan 12
device(config-vlan-12)# untagged ethernet 1/1/5 to 1/1/8
device(config-vlan-12)# tagged ethernet 1/1/23 to 1/1/24
device(config-vlan-12)# source-guard enable ethernet 1/1/23 to 1/1/24
```

The following example configures IPSG on a single port on a VLAN.

```
device# configure terminal
device(config)# vlan 12
device(config-vlan-12)# untagged ethernet 1/1/5 to 1/1/8
device(config-vlan-12)# tagged ethernet 1/1/23 to 1/1/24
device(config-vlan-12)# source-guard enable ethernet 1/1/23
```

The following example configures IPSG on all ports on a VLAN.

```
device# configure terminal
device(config)# vlan 12
device(config-vlan-12)# untagged ethernet 1/1/5 to 1/1/8
device(config-vlan-12)# tagged ethernet 1/1/23 to 1/1/24
device(config-vlan-12)# source-guard enable
```

The following example configures IPSG on a range of ports on multiple VLANs.

```
device# configure terminal
device(config)# vlan 100 to 150
device(config-mvlan-100-150)# tagged ethernet 1/1/23 to 1/1/24
device(config-mvlan-100-150)# source-guard enable ethernet 1/1/23 to 1/1/24
```

The following example configures IPSG on all ports on multiple VLANs.

```
device# configure terminal
device(config)# vlan 151 to 200
device(config-mvlan-151-200)# tagged ethernet 1/1/23 to 1/1/24
device(config-mvlan-151-200)# source-guard enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.40a | This command was modified to support enabling IPSG on a range of ports. |
| 08.0.61 | An example was added for configuring IP Source Guard on a VLAN. |

| Release version | Command history |
|---|---|
| 08.0.80 | Support was added for configuring this command on a range of VLANs. |

# source-interface

Configures the source IP address of the Network Time Protocol (NTP) packets.

## Syntax

**source-interface** { **ethernet** *unit/slot/port* | **lag** *lag-id* | **loopback** *num* | **ve** *num* }

**no source-interface** { **ethernet** *unit/slot/port* | **lag** *lag-id* | **loopback** *num* | **ve** *num* }

## Command Default

When the system sends an NTP packet, the source IP address is normally set to the lowest IP address of the interface through which the NTP packet is sent.

## Parameters

**ethernet** *unit/slot/port*

Configures the source IP address for an NTP packet as that of the specified Ethernet interface.

**lag** *lag-id*

Configures the source IP address for an NTP packet as a LAG virtual interface.

**loopback** *num*

Configures the source IP address for an NTP packet as that of the specified loopback interface.

**ve** *num*

Configures the source IP address for an NTP packet as that of the specified Virtual Ethernet interface.

## Modes

NTP configuration mode

## Usage Guidelines

The specified interface will be used for the source address for all packets sent to all destinations. If a source address is to be used for a specific association, use the **source** keyword in the **peer** or **server** command.

**NOTE**
If the source interface is not configured, the lowest IP address in the outgoing interface will be used in the NTP packets.

The **no** form of the command resets the source IP address of the NTP packets as the IP address of the interface through which the NTP packets are sent.

## Examples

The following example configures the source IP address for an NTP packet as that of the specified Ethernet interface.

```
device(config)# ntp
device(config-ntp)# source-interface ethernet 1/1/3
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add **lag** *lag-id* options. |

# source-ip

Sets the source IP address of an ERSPAN mirror.

## Syntax

**source-ip** *ip-addr*

**no source-ip** *ip-addr*

## Command Default

A source IP is not configured for the ERSPAN profile.

## Parameters

*ip-addr*

Specifies the IP address in the format A.B.C.D.

## Modes

Monitor profile mode

## Usage Guidelines

The source IP address can be any IP on the router.

The **no** form of the command removes the IP address from the monitor profile.

## Examples

The following example sets the source IP address in ERSPAN profile 3.

```
device(config)# monitor-profile 3 type ERSPAN
device(config-monitor-profile 3)# source-ip 2.2.2.2
device(config-monitor-profile 3)# exit
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# source-guard enable

Enables IP Source Guard (IPSG) on a port, per-port per-VLAN, a VLAN or range of VLANs, a virtual interface, or on a range of ports.

## Syntax

**source-guard enable** [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]...

**no source-guard enable** [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]...

## Command Default

IPSG is disabled.

## Parameters

**ethernet** *unit/slot/port*
Specifies the Ethernet interface and the interface ID in the unit/slot/port format.

**to** *stackid/slot/port*
Specifies a range of Ethernet interfaces.

**lag** *lag-id*
Specifies the LAG virtual interface.

**to** *lag-id*
Specifies a range of LAG IDs.

## Modes

Interface configuration mode

VLAN configuration mode

## Usage Guidelines

You can enable IPSG on a range of ports within a given slot only. Enabling IPSG across multiple slots is not supported.

The **no** form of the command disables IPSG on the specified interface.

## Examples

The following example enables IPSG on the Ethernet 1/1/4 interface.

```
device# configure terminal
device(config)# interface ethernet 1/1/4
device(config-if-e10000-1/1/4)# source-guard enable
```

The following example enables IPSG on a specific port in a virtual interface.

```
device(config)# vlan 2
device(config-vlan-2)# tagged ethernet 1/1/1
Added tagged port(s) ethernet 1/1/1 to port-vlan 2
device(config-vlan-2)# router-interface ve 2
device(config-vlan-2)# interface ve 2
device(config-vif-2)# source-guard enable ethernet 1/1/1
```

The following example enables IPSG on a range of ports in the same slot.

```
device# configure terminal
device(config)# interface ethernet 1/1/1
device(config-if-e10000-1/1/1)# interface ethernet 1/1/21 to 1/1/25
decice(config-mif-1/1/21-1/1/25)# source-guard enable
```

The following error displays if you try to configure ports across multiple slots.

```
device(config)# interface ethernet 1/1/18 to 2/1/18
Error - cannot configure multi-ports on different slot
```

The following example configures IPSG on a range of ports on a VLAN.

```
device# configure terminal
device(config)# vlan 12
device(config-vlan-12)# untagged ethernet 1/1/5 to 1/1/8
device(config-vlan-12)# tagged ethernet 1/1/23 to 1/1/24
device(config-vlan-12)# source-guard enable ethernet 1/1/23 to 1/1/24
```

The following example configures IPSG on a single port on a VLAN.

```
device# configure terminal
device(config)# vlan 12
device(config-vlan-12)# untagged ethernet 1/1/5 to 1/1/8
device(config-vlan-12)# tagged ethernet 1/1/23 to 1/1/24
device(config-vlan-12)# source-guard enable ethernet 1/1/23
```

The following example configures IPSG on all ports on a VLAN.

```
device# configure terminal
device(config)# vlan 12
device(config-vlan-12)# untagged ethernet 1/1/5 to 1/1/8
device(config-vlan-12)# tagged ethernet 1/1/23 to 1/1/24
device(config-vlan-12)# source-guard enable
```

The following example configures IPSG on a range of ports on multiple VLANs.

```
device# configure terminal
device(config)# vlan 100 to 150
device(config-mvlan-100-150)# tagged ethernet 1/1/23 to 1/1/24
device(config-mvlan-100-150)# source-guard enable ethernet 1/1/23 to 1/1/24
```

The following example configures IPSG on all ports on multiple VLANs.

```
device# configure terminal
device(config)# vlan 151 to 200
device(config-mvlan-151-200)# tagged ethernet 1/1/23 to 1/1/24
device(config-mvlan-151-200)# source-guard enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.40a | This command was modified to support enabling IPSG on a range of ports. |
| 08.0.61 | An example was added for configuring IP Source Guard on a VLAN. |

| Release version | Command history |
|---|---|
| 08.0.80 | Support was added for configuring this command on a range of VLANs. |

# spanning-tree

Configures STP on all ports on a device.

## Syntax

**spanning-tree** [ **single** ] [ **forward-delay** *seconds* ] [ **hello-time** *seconds* ] [ **max-age** *seconds* ] [ **priority** *number* ]

**no spanning-tree** [ **single** ] [ **forward-delay** *seconds* ] [ **hello-time** *seconds* ] [ **max-age** *seconds* ] [ **priority** *number* ]

## Command Default

STP is not enabled. Once STP is enabled, the STP port parameters are preconfigured with default values.

## Parameters

**single**
Enables Single STP.

**forward-delay** *seconds*
Configures the time period a port waits before it forwards an RST BPDU after a topology change. This value ranges from 4 through 30 seconds. The default is 15 seconds.

**hello-time** *seconds*
Configures the time interval between two Hello packets. This value ranges from 1 through 10 seconds. The default is 2 seconds.

**max-age** *seconds*
Configures the time period the device waits to receive a Hello packet before it initiates a topology change. The time period ranges from 6 through 40 seconds. The default is 20 seconds.

**priority** *number*
Configures the priority of the bridge. The value ranges from 0 through 65535. A lower numerical value means the bridge has a higher priority. Thus, the highest priority is 0. The default is 32768.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

You can specify some or all of the parameters on the same command line.

The **single** option which configures a Single STP is available only in the global configuration mode.

The value of **max-age** must be greater than the value of **forward-delay** to ensure that the downstream bridges do not age out faster than the upstream bridges (those bridges that are closer to the root bridge).

Configuring the STP parameters is optional. All parameters have default values. Additionally, all values will be globally applied to all ports on the system or on the port-based VLAN for which they are defined.

The **no** form of the command disables STP.

# Examples

The following example configures a Single STP.

```
device(config)# spanning-tree single
```

The following example configures the STP parameters.

```
device(config)# vlan 200
device(config-vlan-200)# spanning-tree forward-delay 4 hello-time 5 max-age 4 priority 20
```

# spanning-tree 802-1w

Configures the 802.1w parameters.

## Syntax

**spanning-tree 802-1w** [**single** ] [ **force-version** *number*] [ **forward-delay** *seconds* ] [ **hello-time** *seconds* ] [ **max-age** *seconds* ] [ **priority** *number* ]

**no spanning-tree 802-1w** [ **single** ] [ **force-version** *number* ] [ **forward-delay** *seconds* ] [ **hello-time** *seconds* ] [ **max-age** *seconds* ] [ **priority** *number* ]

Interface configuration mode

**spanning-tree 802-1w** { **admin-edge-port** | **admin-pt2pt-mac**}

**no spanning-tree 802-1w** { **admin-edge-port** | **admin-pt2pt-mac**}

## Command Default

The 802.1w port parameters are preconfigured with default values.

## Parameters

**single**

Configures Single STP.

**force-version** *number*

Forces the bridge to send BPDUs in a specific format. 0 for STP compatibility mode and 2 for RSTP default mode.

**forward-delay** *seconds*

Configures the time period a port waits before it forwards an RST BPDU after a topology change. This value ranges from 4 through 30 seconds. The default is 15 seconds.

**hello-time** *seconds*

Configures the time interval between two Hello packets. This value ranges from 1 through 10 seconds. The default is 2 seconds.

**max-age** *seconds*

Configures the time period the device waits to receive a Hello packet before it initiates a topology change. The time period ranges from 6 through 40 seconds. The default is 20 seconds.

**priority** *number*

Configures the priority of the bridge. The value ranges from 0 through 65535. A lower numerical value means the bridge has a higher priority. Thus, the highest priority is 0. The default is 32768.

**admin-edge-port**

Configures the port to be an operational edge port for all VLANs.

**admin-pt2pt-mac**

Configures the port to be on a point-to-point link link for all VLANs.

# Modes

Global configuration mode

VLAN configuration mode

Interface configuration mode

# Usage Guidelines

The value of **max-age** must be greater than the value of **forward-delay** to ensure that the downstream bridges do not age out faster than the upstream bridges (those bridges that are closer to the root bridge).

Configuring the STP parameters is optional. All parameters have default values. Additionally, all values will be globally applied to all ports on the system or on the port-based VLAN for which they are defined.

The **no** form of the command sets the parameters to the default values.

# Examples

The following example shows how to configure the 802.1w parameters.

```
device(config)# vlan 200
device(config-vlan-200)# spanning-tree 802-1w force-version 6 forward-delay 5 hello-time 4 max-age 4
priority 5
```

# spanning-tree 802-1w ethernet

Enables the spanning-tree 802.1w port commands on Ethernet ports.

## Syntax

**spanning-tree 802-1w** [ **single** ] **ethernet** *stackid/slot/port* [ **admin-edge-port** ] [ **admin-pt2pt-mac** ] [ **force-migration-check** ] [ **path-cost** *number* ] [ **priority** *number* ] [ **disable** ]

**no spanning-tree 802-1w** [ **single** ] **ethernet** *stackid/slot/port* [ **admin-edge-port** ] [ **admin-pt2pt-mac** ] [ **force-migration-check** ] [ **path-cost** *number* ] [ **priority** *number* ] [ **disable** ]

## Command Default

The 802.1w port parameters are pre-configured with default values.

## Parameters

**single**

Configures a Single STP.

**ethernet** *stackid/slot/port*

Specifies the Ethernet port on which you want to configure the 802.1w parameters.

**admin-edge-port**

Enables the port as an edge port in the domain.

**admin-pt2pt-mac**

Enables a port that is connected to another port through a point-to-point link. The point-to-point link increases the speed of convergence. This parameter, however, does not auto-detect whether or not the link is a physical point-to-point link.

**force-migration-check**

Forces the specified port to sent one RST BPDU. If only STP BPDUs are received in response to the send RST BPDU, then the port will return to sending STP BPDUs.

**path-cost** *number*

Configures the cost of the port path to the root bridge. 802.1w prefers the path with the lowest cost. The path cost ranges is from 1 through 20,000,000.

**priority** *number*

Sets the priority for the port. The priority value ranges from 0 through 240, in increments of 16. The default value is 128.

**disable**

Disables 802.1w for the interface on the VLAN.

## Modes

Global configuration mode

VLAN configuration mode

# Usage Guidelines

Configuring the parameters is optional. All parameters have default values. Additionally, all values will be globally applied to all ports on the system or on the port-based VLAN for which they are defined.

The **no** form of the command disables the spanning tree on a VLAN

# Examples

The following example shows the spanning tree configuration for the specified Ethernet port.

```
device(config)# vlan 200
device(config-vlan-200)# spanning-tree 802-1w ethernet 1/1/3 admin-edge-port admin-pt2pt-mac force-
migration-check path-cost 5 priority 10
```

# spanning-tree (Ethernet, LAG)

Configures the Spanning Tree Protocol (STP) path and priority costs for an Ethernet port or Link Aggregation Group (LAG).

## Syntax

**spanning-tree** [ **single** ] [ **ethernet** *unit/slot/port* | **lag** *lag-id* ] { **disable** | **path-cost** { *number* | **auto** } | **priority** *number* }

**no spanning-tree** [ **single** ] [ **ethernet** *unit/slot/port* | **lag** *lag-id* ] { **disable** | **path-cost** { *number* | **auto** } | **priority** *number* }

## Command Default

The Ethernet port parameters are preconfigured with default values.

## Parameters

**single**
Configures a single STP instance.

**ethernet** *unit/slot/port*
Specifies the Ethernet port to be configured.

**lag** *lag-id*
Specifies the LAG to be configured.

**disable**
Disables STP for the interface on the VLAN.

**path-cost** *number*
Configures the cost of the port path to the root bridge. The range when "short mode" is configured is from 1 through 65535. The range when "long mode" is configured is from 1 through 200000000.

**path-cost auto**
Configures the cost of the port path to be the value set by the system software.

**priority** *number*
Sets the priority for the port. The priority value ranges from 0 through 240, in increments of 16; or in an SPX system stack, from 0 through 192 (in steps of 64). The default value is 128.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

The **single** keyword is available only in global configuration mode.

Configuring STP parameter values is optional. STP prefers the path with the lowest cost. All parameters have default values. Additionally, all values will be globally applied to all ports on the system or on the port-based VLAN for which they are defined.

The STP path cost calculation method depends on the 802.1D standard that is configured for the port. By default, the 802.1D 1998 standard is used for all ports running spanning tree. The 802.1D 2004 defined set of path costs allows for faster port speeds and is configured globally to ensure that all network bridges are running the same set of STP path costs.

The **no** form of the command disables STP on the Ethernet port.

# Examples

The following example configures the STP path cost and priority for an Ethernet port.

```
device(config)# vlan 10
device(config-vlan-10)# spanning-tree ethernet 1/1/5 path-cost 15 priority 64
```

The following example configures the STP path cost when the 802.1D 2004 path cost method is configured.

```
device(config)# spanning-tree path-cost-method long
device(config)# vlan 10
device(config-vlan-10)# spanning-tree ethernet 1/1/6 path-cost 20000
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include the LAG ID option. |
| 08.0.70 | This command was modified to allow path cost values from the IEEE 802.1D 2004 standards. |

# spanning-tree designated-protect

Disallows the designated forwarding state on a port in STP 802.1d or 802.1w.

## Syntax

**spanning-tree designated-protect**

**no spanning-tree designated-protect**

## Command Default

STP (802.1d or 802.1w) can put a port into designated forwarding state.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of this command allows the designated forwarding state on a port in STP 802.1d or 802.1w. If STP tries to put a port into designated forwarding state, the device puts this port into the designated inconsistent STP state. This is effectively equivalent to the listening state in STP in which a port cannot forward any user traffic. When STP no longer marks this port as a designated port, the port is automatically removed from the designated inconsistent state.

**NOTE**
You use this command to enable Designated Protection at the port-level while the designated inconsistent state is a per-STP-instance, per-port state.

**NOTE**
You cannot enable Designated Protection and Root Guard on the same port.

## Examples

The following example disallows the designated forwarding state on interface 1/1/1.

```
device(config)# ethernet interface 1/1/1
device(config-if-e1000-1/1/1)# spanning-tree designated-protect
```

## History

| Release version | Command history |
|---|---|
| 07.3.00g | This command was introduced. |

# spanning-tree path-cost-method

Configures all ports running Spanning Tree Protocol (STP) to alter the path costs to match either the 802.1D 1998 path costs or the 802.1D 2004 path costs.

## Syntax

**spanning-tree path-cost-method** [ **long** | **short** ]

**no spanning-tree path-cost-method** [ **long** | **short** ]

## Command Default

The default path cost method is short (802.1D 1998 path costs).

## Parameters

**long**

Configures all ports running STP to use the 802.1D 2004 path costs.

**short**

Configures all ports running STP to use the 802.1D 1998 path costs.

## Modes

Global configuration mode

## Usage Guidelines

The 802.1D 2004 defined set of path costs allows for faster port speeds and all network bridges must be running the same set of STP path costs.

The **no** form of the command returns all ports running STP to the default 802.1D 1998 path costs.

## Examples

The following example configures all ports running STP to use the 802.1D 2004 path cost method for determining STP path costs.

```
device# configure terminal
device(config)# spanning-tree path-cost-method long
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# spanning-tree root-protect

Configures STP root guard.

## Syntax

**spanning-tree root-protect**

**no spanning-tree root-protect**

## Command Default

Root guard is disabled by default.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables STP root guard.

## Examples

The following example shows how to enable RSTP on a port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# spanning-tree root-protect
```

# spanning-tree rstp

Enables 802.1w Draft 3 in a port-based VLAN.

## Syntax

**spanning-tree** [ **single** ] **rstp**

**no spanning-tree** [ **single** ] **rstp**

## Command Default

RSTP is disabled by default.

## Parameters

**single**

Configures single RSTP on the device.

## Modes

Global configuration mode

VLAN configuration mode

## Usage Guidelines

You must enter the command separately in each port-based VLAN in which you want to run 802.1w Draft 3.

This command does not enable STP. To enable STP, first enter the **spanning-tree** command without the **rstp** parameter. After you enable STP, enter the **spanning-tree rstp** command to enable 802.1w Draft 3.

The **no** form of the command disables RSTP.

## Examples

The following example shows how to enable RSTP on a port.

```
device(config)# vlan 10
device(config-vlan-10)# spanning-tree rstp
```

# speed-duplex

Sets link speed and mode (full or half duplex, or slave or master).

## Syntax

**speed-duplex** { **10-full** | **10-half** | **100-full** | **100-half** | **1000-full** | **1000-full-master** | **1000-full-slave** | **10g-full** | | **10g-full-master** | **10g-full-slave** | **2500-full** | **2500-full-master** | **2500-full-slave** |**5g-full** |**5g-full-master**|**5g-full-slave**|**auto** }

**no speed-duplex**

## Command Default

By default, the speed is auto-negotiated.

## Parameters

**10-full**
10M, full duplex

**10-half**
10M, half duplex

**100-full**
100M, full duplex

**100-half**
100M, half duplex

**1000-full**
1G, full duplex

**1000-full-master**
1G, full duplex, master

**1000-full-slave**
1G, full duplex, slave

**10g-full**
10G, full duplex

**10g-full-master**
10G, full duplex, master

**10g-full-slave**
10G, full duplex, slave

**2500-full**
2.5G, full duplex

**2500-full-master**
2.5G, full duplex, master

**2500-full-slave**
2.5G, full duplex, slave

**5g-full**
5G, full duplex

**5g-full-master**
5g, full duplex, master

**5g-full-slave**
5g, full duplex, slave

**auto**
Auto-negotiation. This is the default.

# Modes

Interface configuration mode

# Usage Guidelines

The Gigabit Ethernet copper ports are designed to auto-sense and auto-negotiate the speed and duplex mode of the connected device. If the attached device does not support this operation, you can manually enter the port speed to operate at either 10, 100, or 1000 Mbps. The default and recommended setting is 10/100/1000 auto-sense.

On FastIron devices, when setting the speed and duplex-mode of an interface to 1000-full, configure one side of the link as master (1000-full-master ) and the other side as slave (1000-full-slave ).

Both ends of the link must be configured to operate at the same speed.

2500 and the 5G speeds are applicable only to Multi-Gigabit ports.

The **no** form of the command restores the default.

Refer *Ruckus FastIron Management Configuration Guide* for more information.

# Examples

The following example changes the port speed of copper interface 1/1/8 on a device from the default of 10/100/1000 auto-sense, to 100 Mbps operating in full-duplex mode.

```
device(config)# interface ethernet 1/1/8
device(config-if-e1000-1/1/8)# speed-duplex 100-full
```

# History

| Release version | Command history |
|---|---|
| 8.0.20 | This command was introduced. |
| 8.0.30g | This command was modified to specify that he 1000-full setting mode is not applicable to 1G copper ports on the ICX 7250 and ICX 7450. |
| 8.0.40 | This command was modified to specify that on the ICX 7450-32ZP 2.5G ports, the command works in port pairs only. |

| Release version | Command history |
|---|---|
| 8.0.40a | This command was modified to add support for 100M full-duplex mode on the ICX 7750-48C. |
| 8.0.70 | This command was modified to add support for 5G full on the ICX 7650. |

# spt-threshold

Changes the number of packets the device receives using the RP before switching to the SPT.

## Syntax

**spt-threshold** *num-of-packets*

**no spt-threshold** *num-of-packets*

## Command Default

By default, the device switches from the RP to the SPT after receiving the first packet for a given IPv6 PIM Sparse group.

## Parameters

*num-of-packets*
Specifies the number of packets as a 32-bit integer.

## Modes

Router IPv6 PIM configuration mode

## Usage Guidelines

Each IPv6 PIM Sparse router that is a DR for an IPv6 receiver calculates a short path tree (SPT) towards the source of the IPv6 multicast traffic. The first time the device configured as an IPv6 PIM router receives a packet for an IPv6 group, it sends the packet to the RP for that group, which in turn will forward it to all the intended DRs that have registered with the RP. The first time the device is a recipient, it receives a packet for an IPv6 group and evaluates the shortest path to the source and initiates a switchover to the SPT. Once the device starts receiving data on the SPT, the device proceeds to prune itself from the RPT.

You can change the number of packets the device receives using the RP before switching to using the SPT. If you enter a specific number of packets, the device does not switch over to using the SPT until it has sent the number of packets you specify using the RP.

The **no** form of the command resets the default behavior, that is, the device switches from the RP to the SPT after receiving the first packet for a given IPv6 PIM Sparse group. The device maintains a separate counter for each IPv6 PIM Sparse source-group pair.

## Examples

The following example changes the number of packets the device receives using the RP before switching to the SPT.

```
device(config)# ipv6 router pim
device(config-ipv6-pim-router)# spt-threshold 1000
```

The following example changes the number of packets the device receives using the RP before switching to the SPT for a specified VRF.

```
device(config)# ipv6 router pim vrf blue
device(config-ipv6-pim-router-vrf-blue)# spt-threshold 1000
```

# spx allow-pe-movement

Allows you to move PE units to other CB SPX ports without changing the PE ID or changing any related port configuration.

## Syntax

**spx allow-pe-movement**

**no spx allow-pe-movement**

## Command Default

By default, a PE ID may change when a PE unit is moved.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command disables the feature.

The existing PE ID is used but the **pe-id cb-port** *id1 id2* configuration typically used in PE ID assignment is ignored when **allow-pe-movement** is configured.

All port configuration in the PE still applies.

When you move a PE unit to a new port with **spx allow-pe-movement** configured, the system detaches the port, and protocols register a "port down" event. When the PE joins with the same number on a new SPX port or LAG, the unit is treated as a new PE, and protocols initialize normally.

Ruckus recommends removing **spx allow-pe-movement** configuration once you are finished moving PE units. After removing the configuration, execute the **write memory** command.

## Examples

The following example enables retaining IDs when PE units are moved.

```
device# configure terminal
device(config)# spx allow-pe-movement
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# spx cb-configure

Enters 802.1br control bridge (CB) configuration mode, where SPX ports and LAGs are configured.

## Syntax

**spx cb-configure**

**no spx cb-configure**

## Modes

Router configuration mode.

## Usage Guidelines

The **no** form of the command removes all CB configuration and is available only when **cb-enable** configuration is not present.

The **spx cb-configure** command is available only when the control bridge has been enabled on an eligible router. The router must be an ICX 7750 or an ICX 7650 device.

In CB configuration mode, zero-touch provisioning can be enabled, and zero-touch-ports can be configured. In addition, multicast E-CIDs, SPX ports, SPX LAGs, and PE IDs can be configured. SPX ports and LAGs can be configured in advance or on live links.

## Examples

The following example enables 802.1br CB mode and enters CB configuration mode.

```
device# configure terminal
device(config)# spx cb-enable
Spanning Tree Protocols require a reload. Are you sure? (enter 'y' or 'n'): y
ICX7750-26Q Router(config)# System is now in 802.1br control bridge (CB) mode.
[ System reload follows ]
!
!
device# config terminal
device (config)# spx cb-configure
device(config-spx-cb)# ?
  clear                         Clear table/statistics/keys
  end                           End Configuration level and go to Privileged
                                level
  exit                          Exit current level
  multi-spx-lag                 Configure two lags of a live link
  multi-spx-port                Configure two ports of a live link
  no                            Undo/disable commands
  pe-id                         PE ID assignment provision
  quit                          Exit to User level
  show                          Show system information
  spx-lag                       Configure one CB lag
  spx-port                      Configure one or more CB ports
  write                         Write running configuration to flash or terminal
  zero-touch-enable             actively send probe
  zero-touch-ports              Configure zero touch ports
  <cr>
device(config-spx-cb)#
```

The following command removes the CB configuration. The first attempt is blocked because the CB is enabled. After the CB is disabled, the command is allowed, but a warning is displayed, and you are required to confirm the request.

```
device# configure terminal
device(config)# no spx cb-configure
Error! "no spx cb-config" is not allowed due to "spx cb-enable".
device(config)# no spx cb-enable
System is no longer in 802.1br control bridge (CB) mode.
device(config)# no spx cb-config
Warning! will remove all config in "spx cb-config". Are you sure? (enter 'y' or 'n'):
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.50 | The command was modified to include zero-touch-enable and zero-touch-ports commands. |

# spx cb-enable

Enables a standalone ICX 7750 or ICX 7650 or an ICX 7750 or ICX 7650 stack as an SPX control bridge (CB).

## Syntax

**spx cb-enable**

**no spx cb-enable**

## Command Default

SPX is not enabled by default.

## Modes

Router configuration mode

## Usage Guidelines

The **no** form of the command disables SPX on the device.

Enter the command on the standalone unit or on the active controller of the stack that will become the CB for an SPX domain.

If spanning tree protocol (xSTP) is configured on the prospective CB standalone unit or stack, the command must be followed by a system reload. In this case, you are prompted that a reload is required. If you confirm the command, the reload occurs automatically. The CB units are then reloaded, but not the PE units.

Use the **cb-configure** command to configure SPX ports and LAGs.

## Examples

The following example enables a CB on an ICX 7750 or ICX 7650 router.

```
device# configure terminal
device(config)# spx cb-enable
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# spx interactive-setup

Allows you to configure several options interactively: change existing PE IDs, discover and assign IDs to new PE units, and convert existing or new standalone devices to PE units.

## Syntax

**spx interactive-setup**

## Command Default

None.

## Modes

Privileged EXEC mode.

## Usage Guidelines

You can abort SPX interactive-setup at any time by pressing <CTRL>-C.

SPX interactive-setup will abort afer two minutes of inactivity.

You can use SPX interactive-setup for configuration that is not possible with SPX zero-touch features.

You can use SPX interactive-setup to handle invalid topologies by specifying units to form a valid topology.

If not all units are discovered when you run **spx interactive-setup**, you can run the utility again.

## Examples

The following example shows options available under **spx interactive-setup**.

```
ICX7750-20Q Router# spx interactive-setup
You can abort interactive-setup at any stage by <ctrl-c>
0: quit
1: change PE IDs
2: discover and convert new units (no startup-config flash) to PEs
3: discover and convert existing/new standalone units to PEs
Please type your selection:
```

The following example uses the SPX interactive-setup utility to change two existing PE IDs.

```
ICX7750-48F Router(config)# spx cb-config
ICX7750-48F Router(config-spx-cb)# no zero-touch-enable <-- You cannot run spx interactivesetup
                                                           when zero-touch is enabled.

ICX7750-48F Router(config-spx-cb)# end
ICX7750-48F Router# spx interactive-setup
You can abort spx interactive-setup at any stage by <ctrl-c>
0: quit
1: change PE IDs
2: discover and convert new units (no startup-config flash) to PEs
3: discover and convert existing/new standalone units to PEs
Please type your selection: 1
          +----+        +----+
1/1/3--2/3| 18 |2/5==2/5| 17 |2/4--1/1/16
          +----+        +----+
Type "done" to finish, or a new ID for PE 18 (default 18): 23
Type "done" to finish, or a new ID for PE 17 (default 17): 24
Change IDs: 18->23, 17->24,
Do you want to proceed? (enter 'y' or 'n'): y
T=14m43.4: Sending new IDs to PE(s) 17-18...
power down ports to detach PEs: 1/1/3 1/1/16
power up ports: 1/1/3 1/1/16.
Affected PEs will join with new IDs.
Exit spx interactive-setup, reason: done ID changes
Sica Unit id:24, PoD License Capacity:8
Stack unit 24 Power supply 1 is up
Stack unit 24 Power supply 2 is down
Sica Unit id:23, PoD License Capacity:8
Stack unit 23 Power supply 1 is up
Stack unit 23 Power supply 2 is down
sh spx
T=17m13.2: alone: standalone, D: dynamic cfg, S: static
ID Type Role Mac Address Pri State Comment
1 S ICX7750-48XGF alone cc4e.24d2.2c00 0 local Ready
23 D ICX7250-24 spx-pe cc4e.24dc.e9ce N/A remote Ready
24 D ICX7250-24 spx-pe cc4e.24dc.f166 N/A remote Ready
   +---+
2/1| 1 |2/4
   +---+
          +----+        +----+
1/1/3--2/3| 23 |2/5==2/5| 24 |2/4--1/1/16
          +----+        +----+
```

The following example uses spx interactive-setup to discover and add a connected ring of two PEs (ICX 7250 units).

```
ICX7750-48F Router# configure terminal
ICX7750-48F Router(config)# spx cb-enable
System is now in 802.1br control bridge (CB) mode.
ICX7750-48F Router(config)# spx cb-config
ICX7750-48F Router(config-spx-cb)# spx-port 1/1/3
ICX7750-48F Router(config-spx-cb)# spx-port 1/1/16
ICX7750-48F Router# spx interactive-setup
You can abort spx interactive-setup at any stage by <ctrl-c>
0: quit
1: change PE IDs
2: discover and convert new units (no startup-config flash) to PEs
3: discover and convert existing/new standalone units to PEs
Please type your selection: 2
Probing topology to find new units...
Horizontal bars link to discovered units. Vertical bars link to CB or PEs.
#1: icx7250-24-port-management-module CC4E.24DC.E9CE
#2: icx7250-24-port-management-module CC4E.24DC.F166
1/1/3         1/1/16
  |             |
  |             |
2/3           2/4
+----+        +----+
| 1  |2/5==2/5| 2  |
+----+        +----+

Discovered 1 chain/ring
chain #0: Do you want to select this chain?(enter 'y' or 'n'): y
#1: icx7250-24-port-management CC4E.24DC.E9CE, type an ID (No: 0, default: 17): <-- You can change
the default id,
or just type
enter to use
the default
#2: icx7250-24-port-management CC4E.24DC.F166, type an ID (No: 0, default: 18):
2 unit(s) selected: #1: ID=17, #2: ID=18,
#1 #2
          +----+        +----+
1/1/3--2/3| 17 |2/5==2/5| 18 |2/4--1/1/16
          +----+        +----+
Will produce the above topology. Do you accept it? (enter 'y' or 'n'): y
spx interactive-setup discovers 1 chain(s). valid #=1, selected #=1
Send reload to chain0: #1 CC4E.24DC.E9CE ID=17, D0: 2/3, D1: 2/5 to 2/6 2/8
#2 CC4E.24DC.F166 ID=18, D0: 2/5 to 2/6 2/8, D1: 2/4
Exit spx interactive-setup, reason: 1/1/3 (linking to cc4e.24dc.e9ce) down
Exit spx interactive-setup, reason: 1/1/16 (linking to cc4e.24dc.f166) down
Exit spx interactive-setup, reason: 1/1/3 (linking to cc4e.24dc.e9ce) down
Exit spx interactive-setup, reason: 1/1/16 (linking to cc4e.24dc.f166) down
U18-MSG: PS 1, Internal Power supply is up.
Sica Unit id:18, PoD License Capacity:8
Sica Unit id:17, PoD License Capacity:8
ICX7750-48F Router# show spx
T=42m39.1: alone: standalone, D: dynamic cfg, S: static
ID Type Role Mac Address Pri State Comment
1 S ICX7750-48XGF alone cc4e.24d2.2c00 0 local Ready
17 D ICX7250-24 spx-pe cc4e.24dc.e9ce N/A remote Ready
18 D ICX7250-24 spx-pe cc4e.24dc.f166 N/A remote Ready
   +---+
2/1| 1 |2/4
   +---+
          +----+        +----+
1/1/3--2/3| 17 |2/5==2/5| 18 |2/4--1/1/16
          +----+        +----+
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# spx pe-enable

Enables SPX port extender (PE) mode.

## Syntax

**spx pe-enable**

**no spx pe-enable**

## Command Default

PE mode is not enabled by default.

## Modes

Router configuration mode

## Usage Guidelines

The **no** form of the command disables PE mode and returns the unit to regular mode.

> **NOTE**
> After you enter the no form of the command, even if you have executed the **write memory** command to save the **spx pe-enable** configuration in the PE startup file, the device removes the **spx pe-enable** configuration from the PE startup file immediately after entering regular mode. Then, when you enter **spx pe-enable** and **reload**, the device returns an error because **spx pe-enable** is not configured in the startup file.

A device cannot be enabled as a PE if stacking is enabled on the unit. The command is available only on an ICX 7450 router.

A standalone unit switches between regular and Provisional-PE mode immediately when **spx pe-enable** is configured or removed. When a provisional PE returns to regular mode after the **no spx pe-enable** command is issued, the **spx pe-enable** configuration is immediately removed from the PE startup file if the file exists. This is to prevent accidentally reloading the unit to PE mode.

The PE can be configured once it is enabled; however, the PE configuration is not saved until the **write memory** command is entered, followed by the **reload** command.

# Examples

The following example enables an ICX 7450 standalone unit and moves it to Provisional-PE mode.

```
device# configure terminal
device(config)# spx pe-enable
Enter provisional PE mode. CLI is limited to spx unit 1.
After finishing all configuration, please "write memory" and reload this unit to be a PE.
[Provisional-PE]ICX7450-48F Router(config)# show running-config
Current configuration:
!
ver 08.0.40b1T213
!
spx pe-enable
spx unit 1
  module 1 icx7450-48f-sf-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
  spx-port 1/2/1
  spx-port 1/2/3
!
end

[Provisional-PE]device(config)# write memory
Flash Memory Write (118 bytes)
[Provisional-PE]device(config)#
Write spx_pe.boot done.

[Provisional-PE]device(config)# exit
[Provisional-PE]device# reload
Are you sure? (enter 'y' or 'n'): y
```

The following example shows a provisional PE that returns to regular mode after the **no spx pe-enable** is entered.

```
device# configure terminal
device(config)# spx pe-enable
Enter provisional PE mode. CLI is limited to spx unit 1.
After finishing all configuration, please "write memory" and reload this unit to be a PE.
[Provisional-PE]device(config)# write memory
Flash Memory Write (118 bytes)
[Provisional-PE]device(config)#
Write PE startup file done.

[Provisional-PE]device(config)# no spx pe-enable
Leave provisional PE mode. Spx unit 1 configuration becomes invisible.
device(config)#

[ Note: the device immediately removes “spx pe-enable” from the PE startup file. ]

device(config)# spx pe-enable
Enter provisional PE mode. CLI is limited to spx unit 1.
After finishing all configuration, please "write memory" and reload this unit to be a PE.
[Provisional-PE]ICX7450-48F Router(config)# end
[Provisional-PE]ICX7450-48F Router# reload
Error! provisional PE can only reload to PE mode, but boot file has no "spx pe-enable".
Please do "write memory" and try again, or "no spx pe-enable" to go back to switch/router mode.
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# spx ping

Pings specified PE data port, based on its ECID, to determine if the port is reachable.

## Syntax

**spx ping** { *unit* **/** *slot* **/** *port* }

## Parameters

*unit* **/** *slot* **/** *port*
Port to be pinged.

## Modes

Privileged EXEC mode

## Usage Guidelines

ECID pings can be initiated only from the CB.

ECID pings are not supported for SPX ports.

ECID pings do not work for ports that are physically down.

ECID pings place the port under test in loopback and, as a result, can disrupt control and data traffic on the port.

Each ECID ping is sent to a specific PE, and only one ping can be sent at a time.

If an ECID port ping succeeds, the cause of traffic loss on the port is likely related to an application issue, such as incorrect IP settings. If the ECID port ping fails, the cause of traffic loss on the port is likely related to an SPX infrastructure issue.

**show spx csp events distributed**

## Examples

The following example shows a successful test on PE port 17/1/1. Traffic loss on the port may be due to a problem with IP settings or other applications issues.

```
ICX7750-48F Router# spx ping 17/1/1
SPX Ping Port is disruptive to control, data traffic. Are you sure, you want to continue (enter 'y' or
'n'): y

ICX7750-48F Router# Received response (seq# 6) for ecid1 ping to 17/1/1 port from PE 17
```

The following example shows a failed test on PE port 17/1/2. The port cannot be reached, possibly due to an SPX infrastructure issue.

```
ICX7750-48F Router# spx ping 17/1/2
SPX Ping Port is disruptive to data traffic. Are you sure, you want to continue (enter 'y' or 'n'): y


ICX7750-48F Router# No ecid ping response for spx port 17/1/2, seq 8 from PE 17 !
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# spx suggested-id

Defines a preferred ID for the PE unit being configured.

## Syntax

**spx suggested-id** *number*

## Command Default

The provisional PE does not suggest an ID by default.

## Parameters

*number*

Decimal number from 17 through 56 proposed as the ID for the PE unit being configured.

## Modes

Provisional-PE configuration mode

PE configuration mode

## Usage Guidelines

The command is available only on the ICX 7450, and only after the **spx pe-enable** command has been entered. That is, the command is available only on a PE or a provisional PE unit; it is not available from the control bridge (CB).

The suggested ID does not necessarily become the PE ID. Reserved configuration on the CB that matches the new PE unit takes precedence. Furthermore, if the ID is already assigned, it is not reassigned.

All SPX configuration created on the provisional PE must be saved with the **write-memory** command, followed by the **reload** command to take effect.

## Examples

The following example suggests that the PE being configured be given the PE ID 20 when it joins a CB. Locally, the provisional PE or PE always refers to itself as SPX unit 1.

```
device# configure terminal
device(config)# spx pe-enable
Enter provisional PE mode. CLI is limited to spx unit 1.
After finishing all configuration, please "write memory" and reload this unit to be a PE.
[Provisional-PE]device(config)# spx suggested-id 20
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# spx unconfigure

Removes the PE startup file and recovers the designated PE unit or units to regular mode.

## Syntax

**spx unconfigure** { **me** | *id* | **all** | **unit-id** *id-list* }

## Parameters

**me**

Unconfigures the PE on which the command is issued.

*id*

Entered as a decimal from a CB, unconfigures the PE unit with that ID.

**all**

If issued from a CB, unconfigures all PEs in the SPX domain.

**unit-id** *id-list*

If issued from a CB, unconfigures multiple PEs designated in the *id-list*. The list of units is separated by commas. No spaces are allowed in the list. A range may also be included; for example:

- **spx unconfigure unti-id** *17-19*
- **spx unconfigure unit-id** *17,18,19-21,23*

## Modes

Provisional-PE mode

PE mode

CB router mode

## Usage Guidelines

The **spx unconfigure all** command can be issued only on a CB. This form of the command removes the SPX startup file of every PE and CB unit in the SPX domain. It also reloads all PE units. The CB unit from which the command is issued and other CB units in the configuration are not reloaded.

The **spx unconfigure** *id* command can be issued only from a CB. It removes the PE startup file of the specified PE and reloads it.

The **spx unconfigure unit-id** *id-list* command can be issued only from a CB. It removes the PE startup file of multiple specified PEs and reloads them.

The **spx unconfigure me** command removes the PE startup file from the PE unit on which the command is issued. The startup configuration file for regular mode is not affected. If it is a PE unit, it reloads. A device in regular mode or in Provisional-PE mode does not reload.

The startup configuration file for regular mode is not affected by the **spx unconfigure** command.

# Examples

In the following example, the **spx unconfigure me** command is entered on a PE unit.

```
[PE]local-id@device# spx unconfigure me
This unit will remove the PE startup file and reload as a standalone. Are you sure? (enter 'y' or 'n'):
```

The following example shows the system response when the **spx unconfigure me** command is entered in Provisional-PE mode (on a PE that is configured but for which the configuration has not been written to memory and reloaded).

```
[Provisional-PE]device# spx unconfigure me
This unit will remove the PE startup file. Are you sure? (enter 'y' or 'n'):
```

The following example unconfigures a list of PE units from the CB. No spaces are allowed in the PE list.

```
ICX 7750# configure terminal
ICX 7750(config)# spx unconfigure unit-id 17,18,19-22,25
```

# History

| Release version | Command history |
|---|---|
| 08.0.40 | This command was introduced. |
| 08.0.50 | The **unit-id** key word was added to allow for a list of IDs to be unconfigured simultaneously. |

# spx unit

Configures a reserved SPX unit or certain parameters on a live SPX unit.

## Syntax

**spx unit** *id*

**no spx unit** *id*

## Command Default

Without prior configuration, the SPX unit joins an SPX domain as a dynamic PE, and its dynamic configuration is lost when it leaves.

## Parameters

*id*

Designates the PE unit to be configured.

## Modes

CB configuration mode

PE configuration mode

Provisional-PE configuration mode

## Usage Guidelines

The **no** form of the command removes a reserved SPX unit. It can be entered only from a CB unit. An error message is displayed if the **no** form of the command is entered for a live unit, and the command fails. As an exception, you can remove a live unit that is in a non-operational state due to configuration (module) mismatch. In this case, the CB can remove the configuration, learn the modules from the PE unit, and place it in a ready state.

A CB unit can configure an SPX unit with a unit number from 17 through 56. A PE or provisional PE can only configure unit 1 (itself) and does not have knowledge of its future ID number in the SPX domain.

These command can be configured under **spx unit**: **spx-name**, **module**, **spx-port**, **spx-lag**. On a PE or provisional PE, the **module** command cannot be configured.

If you configure parameters for a live SPX unit, the CB pushes the new configuration to the live unit immediately.

When you create a reserved SPX unit, you must configure modules for the unit. Modules can only be configured on a CB unit.

# Examples

The following example creates a configuration for SPX unit 18 from a CB.

```
device# configure terminal
device(config)# spx unit 18
device(config-spx-unit-18)# spx-name bldg2-floor2-stk 18
device(config-spx-unit-18)# module 1 icx7450-48f-sf-port-management-module
device(config-spx-unit-18)# module 2 icx7400-xgf-4port-40g-module
device(config-spx-unit-18)# module 4 icx7400-qsfp-1port-40g-module
```

The following example configures an SPX port and an SPX LAG from a CB.

```
device# configure terminal
device(config)# spx unit 21
device(config-spx-unit-21)# spx-port 21/2/4
device(config-spx-unit-21)# spx-lag 21/1/10 to 21/1/11
device(config-spx-unit-21)# exit
```

The following example first shows a PE being created, with system-generated ports and module information. The SPX number is always 1 in PE or Provisional-PE configuration mode. SPX ports 1/2/1 and 1/2/3 are generated by the system in this case because there is a 4 X 10-Gbps module in slot 2. The example then shows a user modification to a LAG and an SPX port, still in Provisional-PE mode. As indicated in system messages, you should use the **write memory** command to save the configuration.

```
device(config)# spx pe-enable
Enter provisional PE mode. CLI is limited to spx unit 1.
After finishing all configuration, please "write memory" and reload this unit to be a PE.
[Provisional-PE]device(config)# show run
Current configuration:
!
ver 08.0.40b1T213
!
spx pe-enable
spx unit 1
  module 1 icx7450-48f-sf-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
  spx-port 1/2/1
  spx-port 1/2/3
!
end

[Provisional-PE]device(config)# spx unit 1
[Provisional-PE]device(config-spx-unit-1)# spx-lag 1/2/1 to 1/2/2


spx-port 1/2/1 is replaced by spx-lag 1/2/1 to 1/2/2.
[Provisional-PE]device(config-spx-unit-1)# no spx-port 1/2/3
spx-port 1/2/3 is removed
[Provisional-PE]device(config-spx-unit-1)# spx-port 1/2/4
[Provisional-PE]device(config-spx-unit-1)# show run
Current configuration:
!
ver 08.0.40b1T213
!
spx pe-enable
spx unit 1
  module 1 icx7450-48f-sf-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
  spx-lag 1/2/1 to 1/2/2
  spx-port 1/2/4
!
end
[Provisional-PE]device# write memory
Flash Memory Write (124 bytes)
[Provisional-PE]device#
Write PE startup file done.

[Provisional-PE]device# show configuration
Configuration in PE startup file:
!
ver 08.0.40b1T213
!
spx pe-enable
spx unit 1
  module 1 icx7450-48f-sf-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
  spx-lag 1/2/1 to 1/2/2
  spx-port 1/2/4
!
[Provisional-PE]device# reload
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# spx zero-touch-deny

Configures a standalone unit so that it cannot be discovered by the SPX zero-touch or SPX interactive-setup utility.

## Syntax

**spx zero-touch-deny**

**no spx zero-touch-deny**

## Command Default

By default, an eligible unit is discoverable (within a presentable configuration).

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command returns the unit to a discoverable state. It does not re-enable the zero-touch feature, however.

The **spx zero-touch-deny** command also removes **zero-touch-enable** and **spx pe-enable** under SPX CB configuration.

The command can be entered from an ICX 7250 or an ICX 7450 standalone.

The command cannot be applied to an ICX 7250 or ICX 7450 that is already in Provisional-PE or PE mode.

If you configure **spx pe-enable** on a device, the **spx zero-touch-deny** configuration is removed. Likewise, the spx zero-touch-deny command removes the **spx pe-enable** command.

## Examples

The following example configures the unit so that it cannot be discovered as an SPX PE candidate.

```
device# configure terminal
device(config)# spx zero-touch-deny
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# spx-lag

Configures one end of a multi-port connection on a CB or a PE unit.

## Syntax

**spx-lag** *port-list* [ **pe-group** *name* ]

**no spx-lag** *port-list* [ **pe-group** *name* ]

## Command Default

By default, a LAG does not exist.

## Parameters

*port-list*

Designates the ports to include in the LAG. The port list can contain a list of ports (1/1/2 2/1/2 3/1/2), a range of ports (1/1/2 to 1/1/3), or a combination (1/1/2 to 1/1/3 3/1/2).

**pe-group** *name*

Designates the PE group name associated with the LAG. This option is available in CB configuration mode.

## Modes

CB configuration mode

Provisional-PE mode

PE mode

## Usage Guidelines

The **no** form of the command with the correct list of ports in the LAG removes the LAG. The optional **pe-group** configuration is ignored when removing the SPX LAG.

An SPX LAG can contain from 2 through 16 ports. An SPX LAG allows noncontiguous ports. The **spx-lag** command can be configured for a control bridge (CB) as a cascade LAG or for a PE unit. An SPX LAG on a CB can span multiple units. An SPX LAG on a PE unit can contain only ports on the same unit.

You can remove a LAG member by re-entering the **spx-lag** command without the port number that you want to remove. If you create a new SPX LAG that contains a member of another SPX LAG or a previously configured SPX port, the new LAG replaces the old LAG or SPX port.

SPX LAGs and SPX ports are mutually exclusive in their membership. For example, if you configure an SPX LAG containing ports 17/2/1 to 17/2/3, the system removes the configured SPX port 17/2/1. If you configure SPX port 17/2/1, the system removes the port from the LAG and creates the two-port LAG 17/2/2 to 17/2/3.

The system blocks the **spx-lag** command if executing it would make any PE unreachable. However, it allows the command to break a ring into two chains.

The optional PE group name can be used when assigning PE IDs from the CB with the **pe-id** command. It can also be used in the **show spx pe-group** command to focus command output. If you enter the **spx-lag** command and omit the previously assigned PE group name, the name is removed from the LAG, and it is replaced by the primary port of the SPX LAG in the SPX configuration.

# Examples

The following example configures a LAG that includes ports from CB units 1, 2, and 3.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# spx-lag 1/1/10 2/1/2 3/1/1
```

The following example configures a LAG and then configures a second LAG that replaces the first because port 2/1/9 is in both LAGs.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# spx-lag 1/1/7 2/1/9 3/1/10
device(config-spx-cb)# spx-lag 2/1/9 3/1/12
spx-lag 1/1/7 2/1/9 3/1/10  is replaced by spx-lag 2/1/9 3/1/12.
```

The following example configures a two-port SPX LAG on a provisional PE.

```
device# configure terminal
device(config)# spx pe-enable
Enter provisional PE mode. CLI is limited to spx unit 1.
After finishing all configuration, please "write memory" and reload this unit to be a PE.
[Provisional-PE]device(config)#
[Provisional-PE]device(config)# spx unit 1
[Provisional-PE]device(config-spx-unit-1)# spx-lag 1/2/1 to 1/2/2
```

The following example is executed from the CB and removes an SPX LAG from a PE unit. The command can be entered from the CB for a live PE or for a reserved configuration.

```
ICX7750-20Q Router# configure terminal
ICX7750-20Q Router(config)# spx unit 17
ICX7750-20Q Router(config-spx-unit-17)# no spx-lag 17/2/1 to 17/2/2

spx-lag 17/2/1 to 17/2/2 is removed
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# spx-mon enable

Enables spx-mon analysis tools.

## Syntax

**spx-mon enable**

**no spx-mon enable**

## Command Default

The tool is disabled by default.

## Modes

Global configuration mode.

## Usage Guidelines

The **no** form of the command disables spx-mon.

## Examples

The following example enables spx-mon tools on the device.

```
device# configure terminal
device(conf)# spx-mon enable
```

## History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# spx-port

Configures one or more SPX ports for the control bridge (CB) or a port extender (PE) unit.

## Syntax

**spx-port** *unit/slot/port* [ **pe-group** *name* ]

**no spx-port** *unit/slot/port* [ **pe-group** *name* ]

## Command Default

By default, no SPX port is configured. Some exceptions are noted under *Usage Guidelines*.

## Parameters

*unit/slot/port*
Designates the port to be configured as an SPX port.

**pe-group** *name*
Names the PE group associated with the SPX port or LAG.

## Modes

CB configuration mode

SPX configuration mode (on the CB)

Provisional-PE configuration mode

PE configuration mode

## Usage Guidelines

The **no** form of the command removes the SPX port.

The **pe-group** option is available only for CB SPX ports or LAGs, not for SPX ports or LAGs on a PE unit.

SPX LAGs and SPX ports are mutually exclusive in their membership. For example, if you configure an SPX LAG containing ports 17/2/1 to 17/2/3, the system removes the configured SPX port 17/2/1. If you configure SPX port 17/2/1, the system removes the port from the SPX LAG and creates the two-port LAG 17/2/2 to 17/2/3.

The system blocks the **spx-port** command if executing it would make any PE unreachable.

The **pe-group** *name* option can be used in the **pe-id** command when assigning PE IDs. It can also be used in the **show spx pe-group** command to focus command output.

If you enter the **spx-port** command and omit the previously assigned pe-group name, the name is removed from the port, and it is replaced by the port number in display output.

When you configure **spx pe-enable** on a device and it enters Provisional-PE mode the first time, the device generates two SPX ports. If it has any 4 X 10-Gbps modules installed, the system generates SPX port 1/x/1 and SPX port 1/x/3, where "x" represents the lowest module number of any 4 X 10-Gbps modules installed. If no 4 X 10-Gbps module is

installed, the device generates up to two SPX ports using installed 40-Gbps modules. If no 4 X 10-Gbps or 40-Gbps module is installed in the device, no SPX ports are generated.

# Examples

The following example configures SPX port 1/2/2 on the CB.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# spx-port 1/2/2
```

In the following example, the CB configures SPX port 21/2/4 as part of the reserved configuration for PE unit 21.

```
device# configure terminal
device(config)# spx unit 21
device(config-spx-unit-21)# spx-port 21/2/4
device(config-spx-unit-21)# exit
device(config)#
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# ssh

Starts an SSH2 client connection to an SSH2 server using password authentication.

## Syntax

**ssh** { *hostname* | *ipv4-address* } [ **public-key** { **dsa** | **rsa** } ] [ *port-num* ]

**ssh ipv6** { *hostname* | *ipv6-address* } [ **public-key** { **dsa** | **rsa** } ] [ **outgoing-interface** *type number* ] [ *port-num* ]

## Command Default

SSH2 client connection is not established.

## Parameters

*hostname*
Specifies the host name of the SSH server.

*ipv4-address*
Specifies the IPv4 address of the SSH server.

**public-key**
Configures the type of public key authentication to use for the connection. If you do not enter this parameter, the default authentication type is password.

**dsa**
Specifies the public key authentication type as DSA.

**rsa**
Specifies the public key authentication type as RSA.

*port-num*
Specifies that the SSH2 connection will use a non-default SSH2 port. The default is 22.

**ipv6**
Identifies the remote IPv6 SSH server.

*ipv6-address*
Specifies the IPv6 address of the SSH server.

**outgoing-interface**
Configures the outgoing interface for Link-Local address.

*type*
Specifies the interface type.

*number*
Specifies the interface number. Use **?** to get the list of supported interfaces.

## Modes

Privileged EXEC mode

# Examples

The following example starts an SSH2 client connection to an SSH2 server using password authentication.

```
device# ssh 192.168.10.1
```

The following example starts an SSH2 client connection to an SSH2 server using public key authentication.

```
device# ssh ipv6 2001::1 public-key dsa
```

The following example starts an SSH2 client connection to an SSH2 server using public key authentication.

```
device# ssh ipv6 2001::1 public-key dsa outgoing-interface ethernet 1/1/1 26
```

# ssh access-group

Configures an ACL that restricts SSH access to the device.

## Syntax

**ssh access-group** { *acl-num* | *acl-name* | **ipv6** *ipv6-acl-name* }

**no ssh access-group** { *acl-num* | *acl-name* | **ipv6** *ipv6-acl-name* }

## Command Default

SSH access is not restricted.

## Parameters

*acl-num*
The standard access list number. The valid values are from 1 through 99.

*acl-name*
The standard access list name.

**ipv6** *ipv6-acl-name*
The IPv6 access list name.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the SSH access restriction.

## Examples

The following example shows how to configure an ACL that restricts SSH access to the device. In this example, ACL 10 is configured. The device allows SSH access to all IP addresses except those listed in ACL 10.

```
device(config)# access-list 10 permit host 10.168.144.241
device(config)# access-list 10 deny host 10.168.144.242 log
device(config)# access-list 10 permit host 10.168.144.243
device(config)# access-list 10 deny any
device(config)# ssh access-group 10
```

# ssm-enable

Globally enables source-specific multicast (SSM).

## Syntax

**ssm-enable** [ **range** { *group-address address-mask* | *acl-id* } ]

**no ssm-enable range** { *group-address address-mask* | *acl-id* }

## Command Default

SSM mode is disabled.

## Parameters

**range**

Configures the IP multicast address range.

*group-address address-mask*

Specifies the IP multicast group address and network mask. If this is not configured, the range willdefault to 232/8 as assigned by the Internet Assigned Numbers Authority (IANA) for use with SSM

*acl-id*

Specifies the ACL number or name.

## Modes

IPv4 PIM router configuration mode

IPV6 PIM router configuration mode

IPv4 PIM router configuration mode VRF configuration mode

IPv6 PIM router configuration modeVRF configuration mode

## Usage Guidelines

PIM-SM must be enabled on any ports on which you want SSM to operate.

In the case of IPv4 PIM router configuration mode, the *address-range* can be specified in the format A.B.C.D P.Q.R. S where P.Q.R.S is the network mask or as A.B.C.D/L. If the address is not configured, the range will default to 232/8 as assigned by the Internet Assigned Numbers Authority (IANA) for use with SSM.

In the case of IPv6 PIM router configuration mode, the *address-range* can be specified in the format X:X::X:X/M. If the address is not configured, the address range will default to ff30:/12 as assigned by the Internet Assigned Numbers Authority (IANA) for use with SSM.

The **no** form of the command restores the default.

# Examples

The following example enables SSM on an IPv6 PIM-SM-enabled port.

```
device# configure terminal
device(config)# ipv6 router pim
device(config-ipv6-pim-router)# ssm-enable
```

The following example enables SSM on an IPv4 PIM-SM-enabled port.

```
device# configure terminal
device(config)# router pim
device(config-pim-router)# ssm-enable
```

The following example configures a single SSM group IP address.

```
device# configure terminal
device(config)# router pim
device(config-pim-router)# ssm-enable range 10.1.1.1/8
```

The following example configures PIM so that it uses the group addresses allowed by ACL “xyz” as its PIM SSM range.

```
device# configure terminal
device(config)# router pim
device(config-pim-router)# ssm-enable range xyz
```

The following example enables SSM on an IP PIM-SM-enabled port for VRF red.

```
device# configure terminal
device(config)# router pim vrf red
device(config-pim-router-vrf-red)# ssm-enable range 10.1.1.1/9
```

# stack disable

Prevents a device from joining a traditional stack and from listening for, or sending, stacking packets.

## Syntax

**stack disable**

**no stack disable**

## Command Default

Stacking is disabled by default.

## Modes

Global configuration mode and Stack unit configuration mode

## Usage Guidelines

To remove the restriction that prevents the unit from joining a stack, use the **no stack disable** command.

## Examples

The following example disables the device from joining a stack.

```
device# configure terminal
device(config)# stack disable
Disable stacking. This unit will not be a part of any stack
```

## History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |

# stack enable

Enables stack configuration on the device. Enter this command on the intended active controller.

## Syntax

**stack enable**

**no stack enable**

## Command Default

Stacking is not enabled on the device.

## Modes

Global configuration mode

Stack unit configuration mode

## Usage Guidelines

Use the **no** form of the command to remove stacking capability from the device.

> **NOTE**
> When you use the **no stack enable** command, the unit can still be called to join an active stack. To prevent this, use the **stack disable** command instead.
> You must remove all configuration information from the port before issuing the **stack enable** command.

For manual configuration, the **stack enable** command must be issued on each device in the stack.

## Examples

The following example enables stack configuration on the device.

```
device# config terminal
device(config)# stack enable
Enable stacking. This unit actively participates in stacking
```

## History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |

# stack mac

Manually configures a specific MAC address for a traditional stack.

## Syntax

**stack mac** *mac-address*

**no stack mac** *mac-address*

## Command Default

Beginning with FastIron release 08.0.20, when a stack is enabled or when hitless-failover occurs, a default stack MAC address is assigned if none is configured. In earlier releases, the stack assumed the MAC address of the active controller by default.

## Parameters

*mac-address*

Specifies the MAC address to be used for the stack.

## Modes

Active stack controller configuration mode

## Usage Guidelines

Enter the **no** form of this command to revert to the use of the active controllers' MAC address.

The MAC address is a hexadecimal value entered in the format xxxx.xxxx.xxxx.

# Examples

The following example configures the stack MAC address manually as 0000.0163.0022.

```
device(config)# stack mac 0000.0163.0022
device(config)# show running-config
Current configuration:
!
ver 08.0.40
!
stack unit 1
  module 1 icx7750-48-xgc-port-management-module
  module 2 icx7750-qsfp-6port-qsfp-240g-module
stack rconsole-off
stack mac 0000.0163.0022
!
breakout ethe 1/2/6
!
!
!
global-stp
!
store-and-forward
!
lag ccep1 dynamic id 3
```

# History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |
| 08.0.20 | Stack behavior was modified so that a default MAC address is assigned when the stack is enabled or when hitless failover occurs if no stack MAC address has been configured. |

# stack secure-setup

Configures a stack automatically, to add units to an existing traditional stack, or to change stack member IDs.

## Syntax

**stack secure-setup**

## Modes

Privileged EXEC mode of a stack unit

## Usage Guidelines

Stacking must be enabled with the **stack enable** command before the **stack secure-setup** command can be issued.

When the **stack secure-setup** command is issued on a unit that is not already the active controller, the unit becomes the active controller.

## Examples

In the following example, an ICX 7250 traditional stack is formed using **stack secure-setup** command.

```
device# stack secure-setup
device# Discovering the stack topology...

Available UPSTREAM units
Hop(s) Id   Type          Mac Address
1       2   ICX7250-24P   cc4e.24b4.7bc0
2       3   ICX7250-24P   cc4e.24b4.7efc
3       4   ICX7250-24    cc4e.24b4.8670
4       5   ICX7250-24    cc4e.24b4.84c0
5       6   ICX7250-24    cc4e.24b4.8064
6       7   ICX7250-24    cc4e.24b4.83a0
7       8   ICX7250-48    cc4e.24b4.2514
8       9   ICX7250-48    cc4e.24b4.2820
9       10  ICX7250-24    cc4e.24b4.8988
10      11  ICX7250-48P   cc4e.24b4.2f28
11      12  ICX7250-48P   cc4e.24b4.2eb0
No new core units found...
Selected Topology:
Active Id   Type          Mac Address
1           ICX7250-24P   cc4e.24b4.7c50
Selected UPSTREAM units
Hop(s) Id   Type          Mac Address
1      2    ICX7250-24P   cc4e.24b4.7bc0
2      3    ICX7250-24P   cc4e.24b4.7efc
3      4    ICX7250-24    cc4e.24b4.8670
4      5    ICX7250-24    cc4e.24b4.84c0
5      6    ICX7250-24    cc4e.24b4.8064
6      7    ICX7250-24    cc4e.24b4.83a0
7      8    ICX7250-48    cc4e.24b4.2514
8      9    ICX7250-48    cc4e.24b4.2820
9      10   ICX7250-24    cc4e.24b4.8988
10     11   ICX7250-48P   cc4e.24b4.2f28
11     12   ICX7250-48P   cc4e.24b4.2eb0
Do you accept the unit id's (y/n)?: y
```

# stack suggested-id

Specifies the preferred stack unit ID for a standalone device before it joins a stack.

## Syntax

**stack suggested-id** *stack-unit*

**no stack suggested-id** *stack-unit*

## Parameters

*stack-unit*

Specifies the numeric stack unit ID.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command removes the stack unit ID.

The **stack suggested-id** command is configured on a standalone device before it joins a stack and becomes a member. The command is not for the active controller. Because the active controller always keeps its bootup ID during stack formation, it does not use the suggested-id value.

The system attempts to assign a bootup ID of a device as its stack unit ID. However, due to timing issues or the possible unavailability of the bootup ID, a device might not get the stack unit ID that you want when the stack is formed. The optional **stack suggested-id** command allows you to specify the stack unit ID for member devices when you are configuring a traditional or mixed stack using the manual configuration method.

## Examples

The following example sets the stack unit ID on a standalone device to 3.

```
device# configure terminal
device(config)# stack suggested-id 3
```

# stack suppress-warning

Stops periodic output of background stack diagnostic reports.

## Syntax

**stack suppress-warning**

**no stack suppress-warning**

## Command Default

By default, background diagnostics are displayed periodically on the active stack controller.

## Modes

Stack active controller configuration mode

## Usage Guidelines

Use the **no** form of the command to restore periodic output of background diagnostic reports.

## Examples

In the following example, background diagnostic reports are turned off for the stack.

```
Device# configure terminal
Device(config)# stack suppress-warning
```

# stack switch-over

Switches active controllers without reloading the stack and without packet loss to services and protocols supported by hitless stacking.

## Syntax

**stack switch-over**

## Command Default

With FastIron release 08.0.20, the **stack switch-over** command is allowed by default. In earlier releases, hitless failover must first be enabled.

## Modes

Global configuration mode on a stack controller

## Usage Guidelines

Use the **stack switch-over** command before reloading or performing maintenance on the currently active controller. Hitless failover must be enabled for the command to be used; otherwise, an error message is issued.

The command cannot be used during stack election or during configuration of a multi-stack-trunk.

A standby controller must exist and must have learned stack protocols for the command to be used. The standby controller must have the same priority as the active controller for the command to be used.

More than 120 seconds must have passed since the previous switchover or failover for the command to be accepted.

## Examples

The following example shows the **stack switch-over** command being entered and the resulting output. You must confirm the switch-over before it can take effect by entering **y** when prompted.

```
device# stack switch-over
Standby unit 8 will become active controller, and unit 1 will become standby
Are you sure? (enter 'y' or 'n'): y
Unit 1 is no longer the active controller
```

## History

| Release version | Command history |
|---|---|
| 08.0.00a | This command was introduced. |
| 08.0.20 | Hitless failover is enabled by default. The **stack switch-over** command is allowed by default as a result. |

# stack unconfigure

Returns a stack member to its pre-stacking configuration or state.

## Syntax

**stack unconfigure** [ *stack-unit* | **all** | **me** | **clean** ]

## Parameters

*stack-unit*

Specifies the numerical ID of a stack member. This option is available on the active controller only.

**all**

Specifies all stack members. This option is available on the active controller only.

**me**

Specifies the stack member from which the command is executed. The command removes the unit from the stack and boots it up as a standalone. When the unit rejoins the stack, its standalone startup-config file is saved in a backup file. This option is available on stack member consoles only.

**clean**

Specifies that the startup configuration be removed from the unit on which the command is executed and that the unit be rebooted as a clean unit. This option is available on stack member consoles only.

## Modes

Privileged EXEC mode

## Usage Guidelines

When a stack unit that did not have an original startup configuration file is unconfigured, it becomes a clean unit. It is possible that this unit could automatically rejoin the stack if its module configuration matches the configuration of the active controller. To prevent this from happening accidentally, disconnect the unit to be unconfigured, and then issue the **stack unconfigure me** command on it.

# Examples

In the following example, stack unit 2 is unconfigured in a traditional stack.

```
device(config)# show stack
alone: standalone, D: dynamic config, S: static config
ID  Type          Role    Mac Address    Pri State  Comment
1 S ICX7250-24    active  0012.f2eb.a900 128 local  Ready
2 S ICX7250-24P   standby 00f0.424f.4243 0   remote Ready
3 S ICX7250-24    member  00e0.5201.0100 0   remote Ready

device# stack unconfigure 2
Will recover pre-stacking startup config of this unit, and reset it. Are you sure?
(enter 'y' or 'n'): y

Stack 2 deletes stack bootup flash and recover startup-config.txt from .old

device# show stack
alone: standalone, D: dynamic config, S: static config
ID  Type           Role    Mac Address    Pri  State     Comment
1 S ICX7250-24     active  0012.f2eb.a900 128  local     Ready
2 S ICX7250-24P    member  0000.0000.0000 0    reserved
3 S ICX7250-24     standby 00e0.5201.0100 0    remote    Ready
```

# History

| Release | Command History |
|---|---|
| 07.4.00 | This command was introduced. |
| 08.0.00a | The **mixed-stack** option was added. The **rollback** option was deprecated. |
| 8.0.40 | Removed the **mixed-stack** option as this is not supported on ICX 7750, ICX 7450, and ICX 7250 devices. |

# stack-port

Selects only one of the two stacking ports as a stacking port, which allows you to use the other port as a data port.

## Syntax

**stack-port** *unit*/*slot*/*port*

**no stack-port**

## Command Default

By default, both default ports serve as stacking ports on a stack unit.

## Parameters

*unit*
: Stack unit ID

*slot*
: Slot or module on the unit where the interface resides.

*port*
: Interface to be configured as the sole stack port on the unit.

## Modes

Stack-unit configuration mode.

## Usage Guidelines

The **no** form of the command restores both default stacking ports on the device.

The **stack-port** command should not be used on a live stack. Use the **multi-stack-port** command on a live stack.

Ports identified in the **stack-port** command must be configured as stacking default ports (refer to **default-ports** command page).

## Examples

The following example configures Port 3/2/1 as the only stacking port on stack unit 3.

```
device# configure terminal
device(config)# stack unit 3
device(config-unit-3)# stack-port 3/2/1
Set only one stacking port 3/2/1
```

# stack-trunk

Configures a stack to form a trunk from contiguous links on one side of a stack connection.

## Syntax

**stack-trunk** *stack-unit*/*slot*/*port* **to** *stack-unit*/*slotnum*/*portnum*

**no stack-trunk** *stack-unit*/*slot*/*port* **to** *stack-unit*/*slot*/*port*

## Parameters

*stack-unit*
: Specifies the stack unit ID.

*slot*
: Specifies the slot number.

*port*
: Specifies the port number in the slot.

## Modes

Stack unit configuration mode

## Usage Guidelines

Use the **no** form of the command to disable the stack trunk configuration.

The **stack-trunk** command must be configured on the stack units on both ends of the trunk. Use this command in a new environment on the first deployment of a stack.

To enable the **stack-trunk** command, the primary port in the trunk must be configured under the **stack-port** command configuration.

Do not use the **stack-trunk** command in a production environment. Use the **multi-stack-trunk** command instead.

## Examples

In the following example, ports 1/2/3 and 1/2/4 are configured as a stacking trunk on stack unit 1.

```
Device# configure terminal
Device(config)# stack unit 1
Device(config-unit-1)# stack-trunk 1/2/3 to 1/2/4
```

# static-mac-address

Configures a static MAC address and assigns the address to the premium queue.

## Syntax

**static-mac-address** *ethernet-mac-address* [ **lag** *lag-id* | **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **priority** *number* ]

**static-mac-address** *ethernet-mac-address* [ **lag** *lag-id* | **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **priority** *number* ]

**static-mac-address** *ethernet-mac-address* **drop**

**no static-mac-address** *ethernet-mac-address* **drop**

## Command Default

By default, all MAC addresses are in the best-effort queue.

## Parameters

*ethernet-mac-address*
Specifies the MAC address of the Ethernet interface.

**lag** *lag-id*
Specifies the LAG virtual interface.

**ethernet** *unit/slot/port*
Specifies the Ethernet interface.

**to**
Specifies the range of Ethernet ports.

**priority** *number*
Configures a priority for the Ethernet MAC address. The values are from 0 through 7.

**drop**
Specifies that packets to and from the designated Ethernet MAC address are to be dropped.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command clears the static MAC address configuration.

## Examples

The following example configures a static MAC address on a range of Ethernet interfaces with priority 7.

```
device(config)# vlan 2
device(config-vlan-2)# static-mac-address 0000.0063.67ff ethernet 1/1/1 to 1/1/6 priority 7
```

The following example configures a VLAN to drop packets with a source or destination MAC address.

```
device(config)# vlan 2
device(config-vlan-2)# static-mac-address 0000.0063.67FF drop
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to add the LAG ID option. |

# static-mac-ip-mapping

Adds the client MAC address mapping to the IP address.

## Syntax

**static-mac-ip-mapping** *ip-address mac-address*

**no static-mac-ip-mapping** *ip-address mac-address*

## Parameters

*ip-address*

Specifies the IP address of the client to be used for mapping.

*mac-address*

Specifies the MAC address of the client to be used for mapping.

## Modes

DHCP server pool configuration mode

## Usage Guidelines

The **no** form of the command removes the client MAC address mapping from the IP address.

## Examples

The following example adds the client MAC address mapping to the IP address.

```
device# configure terminal
device(config)# ip dhcp-server pool cabo
device(config-dhcp-cabo)# static-mac-ip-mapping 10.10.10.29 0010.9400.0005
```

## History

| Release version | Command history |
|---|---|
| 08.0.30mb | This command was introduced. |

# store-and-forward

Resets the switching method for forwarding packets from cut-through to store-and-forward.

## Syntax

**store-and-forward**

**no store-and-forward**

## Command Default

The switching method is cut-through.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default packet-forwarding method to cut-through.

Ethernet devices support two basic switching methods for packet forwarding: store-and-forward and cut-through. The default method on ICX 7750 devices is cut-through. You can configure the **store-and forward** command to change it to store-and-forward.

**NOTE**
You must save the configuration and reload for the change to take effect.

A store-and-forward device does not make a forwarding decision on a data packet until it has received the whole frame and checked its integrity; a cut-through device starts the forwarding process soon after it makes the forwarding decision on an incoming frame that is, it might start forwarding before the entire packet is received. This reduces forwarding latency, especially for longer packets. However, there are many factors to consider when selecting which switching method is best for your environment and in some cases it is desirable to change from the default method and configure a device to store-and-forward.

The following table describes some of the differences in how packets are handled depending on the switching method.

| Feature | Cut-through | Store-and-forward |
|---|---|---|
| Forwarding | Data forwarding starts before an entire packet is received | Device waits for entire packet received before processing. |
| Latency | Low latency, less than 1 micro second. | Higher latency; latency depends on frame size. |
| FCS Errors | FCS errors may be propagated from one device to another. | FCS errors are checked and error packets are discarded in the MAC receive. |
| MTU size | MTU size is validated by MAC receive. Oversize packets are marked as error packets but not dropped in the MAC receive. | MTU size is validated by MAC receive. Oversize packets are dropped at the MAC layer. |

# Examples

This example globally enables **store-and-forward** packet switching and saves the configuration.

```
Device(config)# store-and-forward
Device(config)# write memory
Device(config)# end
```

# History

| Release version | Command history |
|---|---|
| 08.0.10b | This command was introduced. |

# stp-bpdu-guard

Enables STP BPDU Guard on the Ethernet interfaces.

## Syntax

**stp-bpdu-guard**

**no stp-bpdu-guard**

## Command Default

STP BPDU Guard is disabled by default.

## Modes

Interface configuration mode

## Usage Guidelines

When a BPDU Guard-enabled port is disabled by BPDU Guard, the device places the port in the errdisable state and displays a message on the console indicating that the port is errdisabled.

The **no** form of the command disables the STP BPDU Guard on the Ethernet interfaces.

## Examples

The following example shows how to enable the STP BPDU Guard on a port.

```
device(config)# interface ethernet 1/2/1
device(config-if-e1000-1/2/1)# stp-bpdu-guard
```

The following example shows how to enable the STP BPDU Guard on multiple ports.

```
device(config)# interface ethernet 1/1/1 to 1/1/9
device(config-mif-1/1/1-1/1/9)# stp-bpdu-guard
```

# stp-group

Changes the CLI to the STP group configuration level.

## Syntax

**stp-group** *group-id*

**no stp-group** *group-id*

## Parameters

*group-id*

Specifies the STP group ID. The value ranges from 1 through 32.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command exits the STP group configuration level.

## Examples

The following example shows how to change to the STP group configuration level.

```
device(config)# stp-group 1
device(config-stp-group-1)#
```

# stp-protect

Prevents an end station from initiating or participating in STP topology changes.

## Syntax

**stp-protect**

**no stp-protect**

## Command Default

STP protection is disabled by default.

## Modes

Interface configuration mode

## Usage Guidelines

This command causes the port to drop STP BPDUs sent from the device on the other end of the link.

The **no** form of the command disables STP protection on the port.

## Examples

The following example shows how to enable STP protection on a port.

```
device(config)# interface ethernet 1/1/2
device#(config-if-e1000-1/1/2)# stp-protect
```

# summary-address (OSPFv2)

Configures route summarization for redistributed routes for an Autonomous System Boundary Router (ASBR).

## Syntax

**summary-address** *A.B.C.D E.F.G.H*

**no summary-address**

## Command Default

Summary addresses are not configured.

## Parameters

*A.B.C.D E.F.G.H*

IP address and mask for the summary route representing all the redistributed routes in dotted decimal format.

## Modes

OSPF router configuration mode

OSPF VRF router configuration mode

## Usage Guidelines

Use this command to configure an ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by a specified address range. When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the device, no action is taken if the device has already advertised the aggregate route; otherwise the device advertises the aggregate route. If an imported route that falls within a configured address range is removed by the device, no action is taken if there are other imported routes that fall within the same address range; otherwise the aggregate route is flushed.

The device sets the forwarding address of the aggregate route to 0 and sets the tag to 0. If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually. If an external link-state-database-overflow condition occurs, all aggregate routes and other external routes are flushed out of the AS. When the device exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges.This parameter affects only imported, type 5 external routes.

The no form of the command disables route summarization.

# Examples

The following example configures a summary address of 10.1.0.0 with a mask of 10.255.0.0. Summary address 10.1.0.0, includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. For all of these networks, only the address 10.1.0.0 is advertised in external LSAs:

```
device# configure terminal
device(config)# router ospf
device(config-ospf-router)# summary-address 10.1.0.0 10.255.0.0
```

# summary-address (OSPFv3)

Configures route summarization for redistributed routes for an Autonomous System Boundary Router (ASBR).

## Syntax

**summary-address** *IPv6-addr/mask*

**no summary-address**

## Command Default

Summary addresses are not configured.

## Parameters

*A:B:C:D/LEN*

IPv6 address and mask for the summary route representing all the redistributed routes in dotted decimal format.

## Modes

OSPFv3 router configuration mode

OSPFv3 VRF router configuration mode

## Usage Guidelines

Use this command to configure an ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by a specified IPv6 address range. When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the device, no action is taken if the device has already advertised the aggregate route; otherwise the device advertises the aggregate route. If an imported route that falls within a configured address range is removed by the device, no action is taken if there are other imported routes that fall within the same address range; otherwise the aggregate route is flushed.

You can configure up to 32 address ranges.

The device sets the forwarding address of the aggregate route to 0 and sets the tag to 0. If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually. If an external link-state-database-overflow condition occurs, all aggregate routes and other external routes are flushed out of the AS. When the device exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges.

If you use redistribution filters in addition to address ranges, the device applies the redistribution filters to routes first, then applies them to the address ranges.

If you disable redistribution, all the aggregate routes are flushed, along with other imported routes.

This option affects only imported, type 5 external routes. A single type 5 LSA is generated and flooded throughout the AS for multiple external routes.

# Examples

The following example configures a summary address of 2001:db8::/24 for routes redistributed into OSPFv3. The summary prefix 2001:db8::/24 includes addresses 2001:db8::/1 through 2001:db8::/24. Only the address 2001:db8::/24 is advertised in an external link-state advertisement.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# summary-address 2001:db8::/24
```

# supportsave (SCP)

Collects logs from different modules and uploads the logs into a remote SCP server.

## Syntax

**supportsave** [ **all** | **cancel** | **core** | **custom** | **info** | **infra** | **l2** | **l3** | **os** | **platform** | **spx** | **system** | *tag* | [ **unit-id** *number* | *tag* ] ]

**supportsave** [ *ipv4address* ] [ **show** ]

**supportsave** [ **add_cust_cmd_index** { *decimal_value "string"* } ]

**supportsave** [ **del_cust_cmd_index** { **all** *integer* } ]

**supportsave** [ **info disable** | **info enable** | **list_cust_cmd** | **show** ]

## Command Default

The supportsave functionality is not active.

## Parameters

**all**

Sends all information to the remote SCP server.

**cancel**

Cancels the **supportsave** command operation.

**core**

Sends core information to the remote SCP server.

**custom**

Sends custom list of information to the remote SCP server.

**info**

Displays information about the **supportsave** command. If info is enabled, then the collected commands contain additional information like BEGIN, CONTEXT, TIME STAMP, HW/SW INFO, and so on.

**infra**

Sends infrastructure information to the remote SCP server.

**l2**

Sends Layer 2 information to the remote SCP server.

**l3**

Sends Layer 3 information to the remote SCP server.

**os**

Sends Operating System information to the remote SCP server.

**platform**

Sends platform information to the remote SCP server.

**spx**

Sends Sequenced Packet Exchange (SPX) information to the remote SCP server.

**system**
Sends system information to the remote SCP server.

*tag*
Appends a text string to the collected file name on the remote SCP server.

**unit-id** *number*
The unit number can be any ID present in the stack. The unit ID accepts only one integer. The logs are collected from the corresponding unit ID and send it to remote server.

**show**
Displays the amount of percentage executed in the currently executing command process.

*ipv4address*
Designates the IP address for the remote server.

**add_cust_cmd index** *integer*
Adds the given command at the given index in the custom commands list. If there is already a command present at the index passed, then add operation will fail.

*string*
The CLI command which is to be added. There is no default value.

*integer*
Index where the command will be added. Valid range 1 to 32. This is a mandatory parameter, with no default value.

**del_cust_cmd index**
Deletes the given command at the given index in the custom commands list. If there is already a command present at the index passed, then add operation will fail.

**all**
Removes all configured custom commands from the supportsave list.

*integer*
Index where the command will be deleted. Valid range 1 to 32. This is a mandatory parameter, with no default value.

**info disable**
Disables the header to be displayed for all show commands being executed.

**info enable**
Enable the header to be displayed for all show commands being executed.

**list_cust_cmd**
Displays the custom command list.

# Modes

Privileged EXEC mode

# Usage Guidelines

The collected logs are shared with the technical support personnel for investigating issues seen on the device. Once the **supportsave** command is executed, logs are collected and uploaded into the remote SCP server.

Parallel execution of **supportsave** command from two different sessions is not allowed. Parallel execution of **supportsave** command and the **copy tftp** or **copy scp** commands is not allowed.

The **supportsave** command supports IPv4.

A maximum of 32 commands can be added to the custom command list. Commands are not expanded while adding a command to the custom commands list. It is recommended not to add any filters with the commands.

Modifying the custom commands list using **supportsave add_cust_cmd** or **supportsave del_cust_cmd** is not allowed while supportsave data collection is in progress.

Time taken by the **supportsave** commands depends on the commands present in the list and the distance of SCP server.

In order to avoid looping, the **supportsave** command cannot be added to the custom command list. Also, the commands which changes the CLI mode (exit, quit) and commands which restart the router (switchover, reload) are not accepted.

The tag string should be less than 11 characters.

The **supportsave** command uses the outbound SSH session

SCP operations are not allowed while **supportsave** is in progress.

Cancelling the **supportsave** command during the file transfer does not cancel the current file transfer. While cancelling the **supportsave** command, you must wait for the current file transfer to complete before executing the **supportsave** command again.

Supportsave is not High Availability (HA) aware.

The **supportsave** command aborts when the remote server is terminated. Additionally, when the data is collected from the remote unit, and if the corresponding unit is powered off, the **supportsave** command is terminated.

Use the **supportsave cancel** command to stop supportsave opertations.

# Examples

Example of **supportsave** command collecting Layer 3 information.

```
device# supportsave l3 scp 10.xx.xx.104
User name:root
Password:Supportsave started. This operation may take several minutes.
Press "Shift-A" to abort supportsave operation.
asethura#####################################################################
Connecting to remote host......

Sending data (8192 bytes per dot)
.

SCP transfer from device completed

Connection Closed

Supportsave completed in 1 seconds
```

Example of **supportsave** command adding a custom command to the fifth position in the index.

```
device# supportsave add_cust_cmd index 5 "host-max-num 512"
```

Example of **supportsave** command deleting a custom command from the fifth position in the index.

```
device# supportsave del_cust_cmd index 5
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# suppress-acl-seq

Suppresses sequence numbers associated with all access control list (ACL) rules.

## Syntax

**suppress-acl-seq**

**no suppress-acl-seq**

## Modes

ACL policy configuration mode

## Usage Guidelines

Upgrading to FastIron release 08.0.50 or a later release adds default sequence numbers (beginning with 10, incrementing by 10) to all configured ACLs.

Before downgrading to a version prior to 08.0.50, run this command to suppress ACL-rule sequence numbers. Although this command removes sequence numbers, it does not change the current order of rules within each ACL.

The **no** form of the command resets the configuration to display default sequence numbers.

## Examples

The following example suppresses rule sequence numbering.

```
device# configure terminal
device(config)# acl-policy
device(config-acl-policy)# suppress-acl-seq
```

The following example restores default rule sequence numbering.

```
device# configure terminal
device(config-acl-policy)# no suppress-acl-seq
device(config-acl-policy)# show ip access-list 8

Standard IP access list 8: 3 entries
10: permit host 1.1.1.1 log
20: permit 36.10.0.0 0.0.0.255 log
30: deny any log
```

## History

| Release | Command History |
|---|---|
| 08.0.50 | This command was introduced. |

# switch-over-active-role

Activates switchover of the active and standby management modules without any packet loss to the services and protocols that are supported by hitless management.

## Syntax

**switch-over-active-role**

## Command Default

Switchover is not enabled.

## Modes

Privileged EXEC mode

## Usage Guidelines

Hitless failover must be enabled before a hitless switchover can be executed.

If this command is entered when hitless failover is disabled, the following message will appear on the console:

```
Switch-over is not allowed. Reason: hitless-failover not configured.
```

**NOTE**
This command is supported only on FastIron SX devices.

## Examples

The following example switches over to the standby module.

```
device# switch-over-active-role
Are you sure? (enter 'y' or 'n'): y
Running Config data has been changed. Do you want to continue
the switch-over without saving the running config? (enter 'y' or 'n'): n
Please save the running config and try switch-over again
```

# symmetric-flow-control enable

Enables symmetric flow control globally on all full-duplex data ports of a standalone unit or on all full-duplex data ports of a particular unit in a traditional stack.

## Syntax

**symmetric-flow-control enable** [ **unit** *stack-unit* [ *stack-unit* ] ... ]

**no symmetric-flow-control enable** [ **unit** *stack-unit* [ *stack-unit* ] ... ]

## Command Default

Symmetric flow control is disabled, and tail drop mode is enabled.

## Parameters

**unit** *stack-unit*

Specifies one of the units in a stacking system for which symmetric flow control is to be enabled. You can specify up to eight units.

## Modes

Global configuration mode

## Usage Guidelines

Because flow control is enabled by default on all full-duplex ports, these ports always honor received 802.3x Pause frames, whether or not symmetric flow control is enabled.

The **no** form of the command disables symmetric flow control.

## Examples

The following example enables symmetric flow control globally on all full-duplex data ports of a standalone unit.

```
device(config)# symmetric-flow-control enable
```

The following example enables symmetric flow control on all full-duplex data ports of unit 4 in a traditional stack.

```
device(config)# symmetric-flow-control enable unit 4
```

# symmetric-flow-control set

Sets symmetric flow control parameters.

## Syntax

**symmetric-flow-control set** *port-type* { **buffers** *value* [ **unit** *unit-value* ] | **xoff** *num* **xon** *num* }

**no symmetric-flow-control set** *port-type* { **buffers** *value* [ **unit** *unit-value* ] | **xoff** *num* **xon** *num* }

## Command Default

1G: Buffers: 272; XOFF Limit: 91; XON Limit: 75

10G: Buffers: 416; XOFF Limit: 91; XON Limit: 75

## Parameters

*port-type*

Specifies the port type. The port type can be one of the following:

**1**

Sets the buffer limits or XOFF and XON limits for 1G ports.

**2**

Sets the buffer limits or XOFF and XON limits for 10G ports.

**3**

Sets the buffer limits or XOFF and XON limits for 100G ports.

**buffers** *value*

Sets the total buffer limits. The value can range from 64 through 320 for 1G ports and from 64 through 1632 for 10G ports. The default value for 1G ports is 272 and for 10G ports is 416 .

**unit** *unit-value*

Specifies the buffer limit for a stack unit.

**xoff** *num*

Sets the XOFF limit. The minimum value is 60 percent, and the maximum value is 95 percent.

**xon** *num*

Sets the XON limit. The minimum value is 50 percent and the maximum value is 90 percent.

## Modes

Global configuration mode

## Usage Guidelines

Use the **show symmetric** command to view the default or configured buffer limit or XON and XOFF thresholds.

The **no** form of the command deletes the configured symmetric flow control values.

# Examples

The following example changes the thresholds for all 1G ports.

```
device(config)# symmetric-flow-control set 1 xoff 91 xon 75
```

The following example changes the total buffer limit for all 10G ports.

```
device(config)# symmetric-flow-control set 2 buffers 128

Total buffers modified, 1G: 320, 10G: 128
```

# symmetrical-flow-control enable

Enables symmetrical flow control (SFC) globally for priorities.

## Syntax

**symmetrical-flow-control enable** [ **all** ]

**no symmetrical-flow-control enable**

## Command Default

SFC is globally disabled.

## Parameters

**all**

Specifies SFC on all priorities. If you do not specify the **all** keyword, SFC is enabled only on priorities 0-4. This parameter is optional.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this restores the default flow-control settings.

Configuring the **symmetrical-flow-control enable** command enables SFC globally for priorities 0-4 by default and optionally for all priorities (0-7)

By default, the system runs in tail-drop mode, with all ports honoring 802.3x flow control and disabling 802.3x transmit. The **symmetrical-flow-control enable** command enables transmission of 802.3x pause frames.

Configuring the **symmetrical-flow-control enable** command changes priority-to-PG mapping.

You cannot configure the **symmetrical-flow-control enable** command if the **priority-flow-control** command is enabled.

If the **symmetrical-flow-control enable** command is not enabled, you cannot configure the **flow-control generate-only** or the **flow-control both** commands in interface configuration mode.

**NOTE**
In FastIron Release 08.0.20 and later releases, SFC is not supported for ports across stack units in ICX 7750 devices or across stack units or for ports across master and slave packet-processor (pp) devices in ICX7450-48 units.

# Examples

The following example shows how to enable SFC:

```
Device(config)# symmetrical-flow-control enable
```

The following example shows how to enable all priorities to send the IEEE 802.3x pause:

```
Device(config)# symmetrical-flow-control enable all
```

The following example shows how to enable SFC for Generate-only mode:

```
Device(config)# symmetrical-flow-control enable
Device(config)# flow-control generate-only
```

The following example shows how to enable SFC for both Honor and Generate-only mode:

```
Device(config)# symmetrical-flow-control enable
Device(config)# flow-control both
```

# History

| Release version | Command history |
|---|---|
| 8.0.10 | This command was introduced. |

# system-max gre-tunnels

Allocates maximum number of GRE tunnels.

## Syntax

**system-max gre-tunnels** *number*

**no system-max gre-tunnels** *number*

## Command Default

Default number of GRE tunnels is 16.

## Parameters

*number*

Specifies the number of GRE tunnels to allocate. Valid value are 16 to 64. The default value is 16.

## Modes

Privileged EXEC mode

## Usage Guidelines

This configuration determines the interface range that is supported for an interface tunnel. For example, if the system-max value is reduced, it is possible that the configured interfaces may be rejected after a system reload.

The **no** form of the command resets the number of GRE tunnels to 16.

## Examples

The following example allocates 60 GRE tunnels.

```
device# system-max gre-tunnels 60
device(config)# write memory
device(config)# exit
device# reload
```

# system-max hw-traffic-conditioner

Configures the maximum number of traffic policies supported on a Layer 3 device.

## Syntax

**system-max hw-traffic-conditioner** *num*

## Command Default

The default is 992.

## Parameters

*num*

Specifies the maximum number of active traffic policies. Value is 992.

## Modes

Global configuration mode

## Examples

The following example sets the maximum number of active traffic policies to 992.

```
device(config)# system-max hw-traffic-conditioner 992
```

# system-max igmp-snoop-group-addr

Sets the maximum number of IGMP group addresses on a device.

## Syntax

**system-max igmp-snoop-group-addr** *num*

**no system-max igmp-snoop-group-addr**

## Command Default

The default number of IGMP group addresses is supported.

## Parameters

*num*

Specifies the maximum number of IGMP group addresses supported. The range is a value from 256 through 8192. The default for IGMP snooping group addresses is 4096.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum.

The configured number of IGMP group addresses is the upper limit of an expandable database. Client memberships exceeding the group limit are not processed.

The following describes the IGMP group address limits for Ruckus devices:

- ICX 7750 switches support 8192 IGMP group addresses.
- ICX 7750 routers support 6K IGMP group addresses.
- ICX 7250 devices support 8192 IGMP group addresses.
- ICX 7450 devices support 8192 IGMP group addresses.

## Examples

This example sets maximum number of IGMP snooping group addresses to 1600.

```
device(config)# system-max igmp-snoop-group-addr 1600
```

# system-max igmp-snoop-mcache

Configures the maximum number of IGMP snooping cache entries supported on a device.

## Syntax

**system-max igmp-snoop-mcache** *num*

**no system-max igmp-snoop-mcache**

## Command Default

The default number of IGMP snooping cache entries is supported.

## Parameters

*num*

Specifies the maximum number of IGMP snooping cache entries supported. The range is a value from 256 through 8192. The default is 512 entries.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum.

The following describes the IGMP snooping multicast cache (mcache) resource limits for Ruckus devices:

- ICX 7750 switches support 8192 IGMP snooping mcache entries.
- ICX 7750 routers support 6K IGMP snooping mcache entries.
- ICX 7250 devices support 8192 IGMP snooping mcache entries.
- ICX 7450 devices support 8192 IGMP snooping mcache entries.

## Examples

This example shows how to configure the maximum number of IGMP snooping mcache entries supported on the device to 2000.

```
device(config)# system-max igmp-snoop-mcache 2000
```

# system-max ip-route

Increases the capacity of the IP route table.

## Syntax

**system-max ip-route** *number*

**no system-max ip-route** *number*

## Command Default

The default is 12000 for ICX 7250 and ICX 7450 devices and 98304 for ICX 7750 and ICX 7650 devices.

## Parameters

*number*

The maximum number of routes in the IP route table.

## Modes

Global configuration mode

## Usage Guidelines

The supported ranges and defaults for IP routes vary by platform:

| Product | Default number of IP routes | **Supported range** |
|---|---|---|
| ICX 7250 | 12000 | 4096 to 15168 |
| ICX 7450 | 12000 | 4096 to 15168 |
| ICX 7650 | 98304 | 98304 to 131072 |
| ICX 7750 | 98304 | 98304 to 131072 |

You must save the configuration and reload the software to place the system maximum change into effect.

The **no** form of the command resets the values to the default.

## Examples

The following example increases the capacity of the IP route table:

```
device(config)# system-max ip-route 5000
device(config)# write memory
device(config)# exit
device# reload
```

# system-max ip-route-default-vrf

Configures maximum IPv4 routes to be allocated for the default VRF instance.

## Syntax

**system-max ip-route-default-vrf** *number*

**no system-max ip-route-default-vrf** *number*

## Command Default

The default number of IPv4 routes to be allocated for the default VRF instance depends on the platform. Refer to the Usage Guidelines section.

## Parameters

*number*

Specifies the number of IPv4 routes to be allocated for the default VRF instance. Refer to the Usage Guidelines section.

## Modes

Global configuration mode

## Usage Guidelines

The maximum, minimum, and default number of IPv4 routes to be allocated for the default VRF instance.

| Platform | Minumum | Default | Maximum |
|---|---|---|---|
| ICX 7250 | N/A | N/A | N/A |
| ICX 7450 | 1024 | 12000 | 15168 |
| ICX 7650 | 256 | 65536 | 131072 |
| ICX 7750 | 256 | 65536 | 131072 |

The **no** form of the command resets the number of IPv4 routes allocated for the default VRF instance to the default.

## Examples

The following example sets the number of IPv4 routes for the default VRF instance as 13000.

```
device(config)# system-max ip-route-default-vrf 13000
device(config)# write memory
```

# system-max ip-route-vrf

Configures default maximum IPv4 routes to be allocated per user-defined VRF.

## Syntax

**system-max ip-route-vrf** *number*

**no system-max ip-route-vrf** *number*

## Command Default

The default number of the maximum IPv4 routes to be allocated per uesr-defined VRF depends on the platform. Refer to the Usage Guidelines section.

## Parameters

*number*

Specifies the number of maximum IPv4 routes to be allocated per uesr-defined VRF. Refer to the Usage Guidelines section.

## Modes

Global configuration mode

## Usage Guidelines

The maximum, minimum, and the default number of IPv4 routes to be allocated per user-defined VRF depends on the platform.

| Platform | Minimum | Default | Maximum |
|---|---|---|---|
| ICX 7250 | N/A | N/A | N/A |
| ICX 7450 | 128 | 1024 | 15168 |
| ICX 7650 | 64 | 4096 | 131072 |
| ICX 7750 | 64 | 4096 | 131072 |

The **no** form of the command resets the number of maximum IPv4 routes to be allocated per user-defined VRF to the default.

## Examples

The following example configures the number of IPv4 routes to be allocated per user-defined VRF as 1500.

```
device(config)# system-max ip-route-vrf 1500
device(config)# write memory
```

# system-max ip-vrf

Configures maximum VRF instances supported by the software.

## Syntax

**system-max ip-vrf** *number*

**no system-max ip-vrf** *number*

## Command Default

The default number of VRF instances supported by the software depends on the platform. Refer to the Usage Guidelines section.

## Parameters

*number*

Configures the number of VRF instances supported. Refer to the Usage Guidelines section.

## Modes

Global configuration mode

## Usage Guidelines

The range of maximum and minimum configurable VRF instance and the default values depends on the platform.

| Platform | Minimum | Default | Maximum |
|---|---|---|---|
| ICX 7250 | 4 | 16 | 16 |
| ICX 7450 | 4 | 16 | 16 |
| ICX 7650 | 16 | 128 | 128 |
| ICX 7750 | 16 | 128 | 128 |

The **no** form of the command resets the VRF instance to the default value.

## Examples

The following example configures the maximum number of VRF instance as 20.

```
device(config)# system-max ip-vrf 20
device(config)# write memory
device(config)# end
```

# system-max ip6-route

Configures maximum IPv6 routes, used to initialize hardware during system init.

## Syntax

**system-max ip6-route** *number*

**no system-max ip6-route** *number*

## Command Default

The default number of routes depends on the platform. Refer to the Usage Guidelines section.

## Parameters

*number*

Specifies the number of IPv6 routes. Refer to the Usage Guidelines section.

## Modes

Global configuration mode

## Usage Guidelines

The maximum and minimum number of IPv6 routes that can be configured depends on the platform.

The **system-max ip6-route** command cannot be configured for ICX7250 and ICX7450 devices, for which system maximum route values are derived from the **ip-route** value.

| Platform | Minimum | Default | Maximum |
|---|---|---|---|
| ICX 7250 | N/A | N/A | N/A |
| ICX 7450 | N/A | N/A | N/A |
| ICX 7650 | 5120 | 5120 | 7168 |
| ICX 7750 | 5120 | 5120 | 7168 |

The **no** form of the command resets the number of IPv6 routes to the default value.

## Examples

The following example configures the number of IPv4 routes as 5000.

```
device(config)# system-max ip6-route 5000
device(config)# write memory
```

# system-max ip6-route-default-vrf

Configures maximum IPv6 routes to be allocated for the default VRF instance.

## Syntax

**system-max ip6-route-default-vrf** *number*

**no system-max ip6-route-default-vrf** *number*

## Command Default

The default number of IPv6 routes to be allocated for the default VRF instance depends on the platform. Refer to the Usage Guidelines section.

## Parameters

*number*

Specifies the number of IPv6 routes to be allocated for the default VRF instance. Refer to the Usage Guidelines section.

## Modes

Global configuration mode

## Usage Guidelines

The maximum, minimum, and default number of IPv6 routes to be allocated for the default VRF instance.

| Platform | Minumum | Default | Maximum |
|---|---|---|---|
| ICX 7250 | N/A | N/A | N/A |
| ICX 7450 | 64 | 5120 | 5120 |
| ICX 7650 | 64 | 2048 | 7168 |
| ICX 7750 | 64 | 2048 | 7168 |

The **no** form of the command resets the number of IPv6 routes allocated for the default VRF instance to the default.

## Examples

The following example sets the number of IPv4 routes for the default VRF instance as 3000.

```
device(config)# system-max ip6-route-default-vrf 3000
device(config)# write memory
```

# system-max ip6-route-vrf

Configures default maximum IPv6 routes to be allocated per user-defined VRF.

## Syntax

**system-max ip6-route-vrf** *number*

**no system-max ip6-route-vrf** *number*

## Command Default

The default number of the maximum IPv6 routes to be allocated per uesr-defined VRF depends on the platform. Refer to the Usage Guidelines section.

## Parameters

*number*

Specifies the number of maximum IPv6 routes to be allocated per uesr-defined VRF. Refer to the Usage Guidelines section.

## Modes

Global configuration mode

## Usage Guidelines

The maximum, minimum, and the default number of IPv6 routes to be allocated per user-defined VRF depends on the platform.

| Platform | Minimum | Default | Maximum |
|---|---|---|---|
| ICX 7250 | N/A | N/A | N/A |
| ICX 7450 | 64 | 100 | 5120 |
| ICX 7650 | 16 | 1024 | 7168 |
| ICX 7750 | 16 | 1024 | 7168 |

The **no** form of the command resets the number of maximum IPv6 routes to be allocated per user-defined VRF to the default.

## Examples

The following example configures the number of IPv6 routes to be allocated per user-defined VRF as 1500.

```
device(config)# system-max ip6-route-vrf 1500
device(config)# write memory
```

# system-max ip-static-arp

Configures the maximum number of static ARP table entries.

## Syntax

**system-max ip-static-arp** *number*

**no system-max ip-static-arp** *number*

## Command Default

The default is 512 entries.

## Parameters

*number*

Specifies the number of entries the static ARP table can hold. Valid range is 512 to 6000. The default is 512.

## Modes

Global configuration mode

## Usage Guidelines

You must save the configuration to the startup-config file and reload the software after changing the static ARP table size to place the change into effect.

The **no** form of the command resets the number of allowable entries the static ARP table to the default value.

## Examples

The following example increases the maximum number of static ARP table entries you can configure to 1000.

```
device(config)# system-max ip-static-arp 1000
device(config)# write memory
device(config)# end
device# reload
```

# system-max ip-subnet-port

Increases the number of IP subnet interfaces that can be configured on each port of the device.

## Syntax

**system-max ip-subnet-port** *number*

**no system-max ip-subnet-port** *number*

## Command Default

The default number of IP subnet interfaces is 24.

## Parameters

*number*

Specifies the maximum number of IP subnets per port. The range is from 24 through 128. The default value is 24.

## Modes

Global configuration mode

## Usage Guidelines

You must save the configuration and reload the software to place the system maximum change into effect.

The **no** form of the command resets the value to the default.

## Examples

The following example increases the capacity of the IP subnet interfaces.

```
device(config)# system-max ip-subnet-port 64
device(config)# write memory
device(config)# exit
device# reload
```

# system-max l3-interface

Configures the maximum number of layer 3 interfaces that can be configured on a system. Applicable for ICX7150 only.

## Syntax

**system-max l3-interface** *num*

**no system-max l3-interface** *num*

## Command Default

By default, 128 layer 3-interfaces can be configured in a system.

## Parameters

*num*

Specifies the maximum number of the layer 3 interfaces that can be configured. Valid values range from 1 through 382. The default is 128.

## Modes

Global configuration mode

## Usage Guidelines

This command is supported for the ICX 7150 only. For other platforms refer to the **system-max virtual-interface** command.

The **no** form of the command removes the configured maximum number of Layer 3 interfaces and resets the maximum value to the default.

## Examples

The following example shows how to increase the maximum number of layer 3 interfaces.

```
device# configure terminal
device(config)# system-max l3-interface 129
device(config)# write memory
device(config)# end
device# reload
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced for the ICX 7150. |

# system-max mac

Changes the capacity of the MAC address table.

## Syntax

**system-max mac** *number*

**no system-max mac** *number*

## Command Default

The default capacity is 32768 MAC addresses.

## Parameters

*number*

The maximum number of MAC addresses in the MAC table. The valid value is 32768.

## Modes

Global configuration mode

## History

| Release | Command History |
|---|---|
| 8.0.40 | The command is not valid as you cannot change the number in ICX 7750, ICX 7450, and ICX 7250 devices. |

# system-max mac-notification-buffer

Changes the value of the MAC-notification buffer.

## Syntax

**system-max mac-notification-buffer** *size*

**no system-max mac-notification-buffer** *size*

## Command Default

The default buffer size is 4000.

## Parameters

***size*** Sets the buffer queue size to maintain MAC-notification events.

## Modes

Global configuration

## Usage Guidelines

The **no** form of the command sets the MAC-notification buffer to default size. The default buffer value is 4000, maximum value is 16000, and the allowed values are 4000, 8000 and 16000.

## Examples

This example changes the value of the MAC-notification buffer:

```
device(config)# system-max mac-notification-buffer 8000
```

This example sets the MAC-notification buffer to default size:

```
device(config)# no system-max mac-notification-buffer 4000
```

## History

| Release version | Command history |
|---|---|
| 08.0.10 | This command was introduced. |

# system-max max-dhcp-snoop-entries

Sets the maximum number of DHCP snooping entries that can be configured for the entire stack.

## Syntax

**system-max max-dhcp-snoop-entries** *number*

**no system-max max-dhcp-snoop-entries** *number*

## Command Default

The default is 8192 entries.

## Parameters

*number*

Specifies the maximum number of DHCP snooping entries that can be learned for the stack. Valid values range from 1024 through 32768. The default is 8192.

## Modes

Global configuration mode

## Usage Guidelines

You must save the configuration and reload the software for the system maximum change to take effect.

The **no** form of the command restores the default value.

## Examples

The following example increases the maximum number of DHCP snooping entries that can be configured to 10000.

```
device# configure terminal
device(config)# system-max max-dhcp-snoop-entries 10000
Reload required.  Please write memory and then reload or power cycle.
device(config)# write memory
device(config)# end
device# reload
```

The following example restores the maximum number of DHCP snooping entries that can be configured to the default value of 8192.

```
device# configure terminal
device(config)# no system-max max-dhcp-snoop-entries
Reload required.  Please write memory and then reload or power cycle.
device(config)# write memory
device(config)# end
device# reload
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was modified to support a maximum of 32768 DHCP snooping entries and change the default value to 8192 entries. |

# system-max max-ecmp

Configures the maximum limit of ECMP paths at the system level.

## Syntax

**system-max max-ecmp** [ *num* ]

**no system-max max-ecmp** [ *num* ]

## Command Default

The default value is 8.

## Parameters

*num*

Specifies the maximum number of ECMP paths and can be from 8 through 32.

## Modes

Global configuration mode

## Usage Guidelines

The **system-max max-ecmp** command is supported only on the Ruckus ICX 7750.

If the maximum number of ECMP paths is not configured at the system level, by default, you can configure the maximum number of IP load sharing paths to a value from 2 through 8.

The configuration of the maximum number of IP load sharing paths to a value more than 8 is determined by the maximum number of ECMP paths configured at the system level using the **system-max max-ecmp** command.

You cannot configure the maximum number of IP load sharing paths higher than the value defined at the system level.

You cannot configure the maximum number of ECMP paths at the system level to a value less than the configured IP load sharing value.

You must save the configuration and reload the device for the maximum ECMP value change to take effect.

The **no** form of the command removes the maximum number of ECMP paths defined at the system level.

## Examples

The following example defines the maximum number of ECMP paths that can be configured in the system as 20.

```
device(config)# system-max max-ecmp 20
device(config)# write memory
device(config)# exit
device# reload
```

# History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# system-max max-ip-mac

Changes the maximum number of MAC addresses that can be configured on IP interfaces.

## Syntax

**system-max max-ip-mac** *number*

**no system-max max-ip-mac** *number*

## Command Default

The default maximum of MAC addresses to be configured on IP interfaces is 120.

## Parameters

*number*

The maximum number of MAC addresses to be configured on IP interfaces. The valid range is from 120 through 248. The default value is 120.

## Modes

Global configuration mode

## Usage Guidelines

Each physical or virtual Ethernet (VE) interface can be configured with only one MAC address. There is a maximum number of IP interfaces (248 on which an IP MAC address can be configured and the number of Virtual Router Redundancy Protocol (VRRP) virtual interfaces that can be supported simultaneously is affected by any increase over the default number of 120 interfaces. If the **system-max max-ip-mac** command is set above 120, a reduction in the number of IPv4 VRRP entries supported is calculated as <configured-value> - 120. For example, if the **system-max max-ip-mac** command is set to 130, the number of IPv4 VRRP entries is reduced by 10 entries (130-120).

You must save the configuration and reload the software before the changed maximum number takes effect.

The **no** form of the command resets the value to the default.

## Examples

The following example increases the maximum number of MAC addresses that can be configured on IP interfaces.

```
device# configure terminal
device(config)# system-max max-ip-mac 140
Total IP-MAC entries supported is changed to 140
 Total VRRP instances supported changed to 370, IPv4 VRRP instances: 228, IPv6 VRRP instances 120
Reload required.  Please write memory and then reload or power cycle.
device(config)# write memory
device(config)# exit
device# reload
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# system-max max-static-inspect-arp-entries

Sets the maximum number of static ARP inspection entries that can be configured for the entire stack.

## Syntax

**system-max max-static-inspect-arp-entries** *number*

**no system-max max-static-inspect-arp-entries** *number*

## Command Default

The default is 512 entries.

## Parameters

*number*

Specifies the maximum number of static ARP inspection entries that can be configured. Valid values range from 512 through 2048. The default is 512.

## Modes

Global configuration mode

## Usage Guidelines

You must save the configuration and reload the software for the system maximum change to take effect.

The **no** form of the command restores the default value.

## Examples

The following example increases the maximum number of static ARP inspection entries that can be configured to 860.

```
device# configure terminal
device(config)# system-max max-static-inspect-arp-entries 860
Reload required.  Please write memory and then reload or power cycle.
device(config)# write memory
device(config)# end
device# reload
```

The following example restores the maximum number of static ARP inspection entries that can be configured to the default value of 512.

```
device# configure terminal
device(config)# no system-max max-static-inspect-arp-entries
Reload required. Please write memory and then reload or power cycle.
device(config)# write memory
device(config)# end
device# reload
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was modified to support a maximum of 2048 static ARP inspection entries. |

# system-max mld-snoop-group-addr

Sets the maximum number of multicast listening discovery (MLD) group addresses on a device.

## Syntax

**system-max mld-snoop-group-addr** *num*

**no system-max mld-snoop-group-addr**

## Command Default

The default number of MLD group addresses is supported.

## Parameters

*num*

Specifies the maximum number of MLD group addresses supported. The range is a value from 256 through 8192. The default for MLD snooping group addresses is 4096.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum.

The configured number of MLD group addresses is the upper limit of an expandable database. Client memberships exceeding the group limit are not processed.

The following describes the MLD group address limits for Ruckus devices:

- ICX 7750 switches support 8192 MLD group addresses.
- ICX 7750 routers support 6K MLD group addresses.
- ICX 7250 devices support 8192 MLD group addresses.
- ICX 7450 devices support 8192 MLD group addresses.

## Examples

This example sets maximum number of MLD snooping group addresses to 4000.

```
device(config)# system-max mld-snoop-group-addr 4000
```

# system-max mld-snoop-mcache

Configures the maximum number of multicast listening discovery (MLD) snooping cache entries supported on a device.

## Syntax

**system-max mld-snoop-mcache** *num*

**no system-max mld-snoop-mcache**

## Command Default

The default number of MLD snooping cache entries is supported.

## Parameters

*num*

Specifies the maximum number of MLD snooping cache entries supported. The range is 256 to 8192. The default is 512 entries.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum.

The following describes the MLD snooping multicast cache (mcache) resource limits for Ruckus devices:

- ICX 7250 and ICX 7450 devices support up to 8192 MLD snooping mcache entries.
- ICX 7750 switches support up to 8192 MLD snooping mcache entries.
- ICX 7750 routers support 3072 MLD snooping mcache entries.
- In Release 8.0.10a and later releases, ICX 7750 routers support 6144 MLD snooping mcache entries.

## Examples

This example shows how to set the maximum number of MLD snooping mcache entries to 8000.

```
device(config)# system-max mld-snoop-mcache 8000
```

# system-max msdp-sa-cache

Configures the maximum number of source active (SA) messages n the Multicast Source Discovery Protocol (MSDP) cache.

## Syntax

**system-max msdp-sa-cache** *num*

**no system-max msdp-sa-cache** *num*

## Command Default

4096 MSDP SA messages is supported.

## Parameters

*num*

Specifies the maximum number of MSDP SA messages supported. The range is 1024 to 8192. The default is 4096 messages.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command restores the default maximum.

## Examples

The following example sets the maximum number of MSDP SA messages to 6000.

```
device(config)# system-max msdp-sa-cache 6000
```

# system-max pim-hw-mcache

Sets the maximum number of SG entries allowed in the device.

## Syntax

**system-max pim-hw-mcache** *num*

**no system-max pim-hw-mcache** *num*

## Command Default

1024 SG entries are supported.

## Parameters

*num*

Specifies the maximum number of entries. The range is 256 to 6144; the default: 1024.

## Modes

Global configuration mode

## Usage Guidelines

The **system max pim-hw-mcache** replace the **system-max pim mcache** command.

The **no** form of the command restores the default maximum.

## Examples

The following example sets the maximum number of SG entries allowed in the device to 900.

```
device(config)# system-max pim-hw-mcache 900
```

# system-max pim6-hw-mcache

Sets the maximum number of SG entries allowed in the device.

## Syntax

**system-max pim6-hw-mcache** *num*

**no system-max pim6-hw-mcache** *num*

## Command Default

512 SG entries are supported.

## Parameters

*num*

Specifies the maximum number of entries. The range is 256 to 1024; the default: 512.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command restores the default maximum.

You can use the **max-mcache** command to define the maximum number of repeated PIM traffic sent from the same source address and received by the same destination address.

## Examples

The following example sets the maximum number of SG entries allowed in the device to 900.

```
device(config)# system-max pim6-hw-mcache 900
```

# system-max pms-global-pool

Configures the maximum number of global resources shared among all interfaces on the device to store secure MAC addresses for port MAC security (PMS).

## Syntax

**system-max pms-global-pool** *num*

**no system-max pms-global-pool** *num*

## Command Default

8192 global resources

## Parameters

*num*

Specifies the number of global resources shared among all interfaces on the device to store secure MAC addresses for PMS. Valid values range from 1 through 8192.

## Modes

Global configuration mode

## Usage Guidelines

The global resources are in addition to the local resources allocated to each interface . The maximum number of MAC addresses any single interface can secure is 64 (the maximum number of local resources allocated to the interface), plus the number of global resources not allocated to other interfaces. Global resources are shared among all the interfaces on a first-come, first-served basis.

The **no** form of the command removes the configured number of global resources and resets the maximum value to the default.

## Examples

The following example sets the maximum number of shared global resource to 800.

```
device# configure terminal
device(config)# system-max pms-global-pool 800
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# system-max rmon-entries

Configures the maximum number of entries allowed in the RMON control table.

## Syntax

**system-max rmon-entries** *value*

**no system-max rmon-entries** *value*

## Command Default

The default number of RMON entries allowed in the RMON control table is 1024 on ICX 7450 and ICX 7250 devices and 2048 on ICX 7750 devices.

## Parameters

*value*

Specifies the maximum number of entries. The value can range from 128 to 32768 on ICX 7450 and ICX 7250 devices. The value can range from 2048 to 32768 on ICX 7750 devices.

## Modes

Global configuration mode

## Usage Guidelines

This command configures the maximum number of entries allowed in the RMON control table, including alarms, history, and events.

**NOTE**
In order for the change to take effect, you must save the change to the startup-config file and reload or reboot.

The **no** form of the command resets the maximum number of entries allowed in the RMON table to the default value.

## Examples

The following example sets the maximum number of RMON entries to 3000.

```
device(config)# system-max rmon-entries 3000
device(config)# write memory
device(config)# exit
device# reload
```

# system-max spanning-tree

Configures the system maximum value for the number of spanning tree instances.

## Syntax

**system-max spanning-tree** *number*

**no system-max spanning-tree** *number*

## Command Default

The default number of spanning tree instances is 32.

## Parameters

*number*

Configures the number of spanning tree instances. The range is from 1 through 254.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the system maximum value of spanning tree instances to the default.

## Examples

The following example shows how to set the maximum number of spanning tree instances.

```
device(config)# system-max spanning-tree 254
```

# system-max view

Configures the maximum number of SNMP views available on a device.

## Syntax

**system-max view** *number-of-views*

**no system-max view** *number-of-views*

## Command Default

The default number of views is 10.

## Parameters

*number-of-views*

Specifies the maximum number of SNMPv2 and SNMPv3 views. The number of views can range from 10 through 65535. The default value is 10.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the number of views to the default value of 10.

## Examples

The following example configures the maximum number of SNMP views as 15.

```
device(config)# system-max view 15
```

# system-max virtual-interface

Increases the maximum number of virtual routing interfaces you can configure.

## Syntax

**system-max virtual-interface** *num*

**no system-max virtual-interface** *num*

## Command Default

The default maximum number of virtual interfaces that can be configured is 255.

## Parameters

*num*

Specifies the maximum number of the virtual routing interface that can be configured. The range depends on the device being configured.

## Modes

Global configuration mode

## Usage Guidelines

The number of virtual routing interfaces supported on your product depends on the device and, for chassis devices, the amount of DRAM on the management module. The **write memory** command must be executed to save the changes and a reload is required.

The **no** form of the command removes the configured maximum number of virtual routing interfaces and resets the maximum value to the default.

## Examples

The following example shows how to increase the maximum number of virtual routing interfaces.

```
device(config)# system-max virtual-interface 512
device(config)# write memory
device(config)# end
device# reload
```

# system-max vlan

Increases the maximum number of VLANs you can configure.

## Syntax

**system-max vlan** *num*

**no system-max vlan** *num*

## Command Default

The default maximum value is 64 VLANs.

## Parameters

*num*

Specifies the maximum number of VLANs you can configure. The range depends on the device being configured.

## Modes

Global configuration mode

## Usage Guidelines

Although you can specify up to 4095 VLANs, you can configure only 4094 VLANs. VLAN ID 4094 is reserved for use by Single STP. The **write memory** command must be executed to save the changes and a reload is required. The number of VLANs supported on your product depends on the device and, for chassis devices, the amount of DRAM on the management module.

The **no** form of the command removes the maximum number of VLANs and resets the maximum value to 64.

## Examples

The following example shows how to increase the maximum number of VLANs.

```
device(config)# system-max vlan 2048
device(config)# write memory
device(config)# end
device# reload
```

# sz active-list

Configures the SmartZone IP addresses that the ICX switch will use first to initiate a connection with SmartZone.

## Syntax

**sz active-list** { *ip-address* } [ *ip-address2* ] [ *ip-address3* ]

**no sz active-list**

## Command Default

SmartZone IP addresses are not configured on the switch, which means the switch will attempt to initiate a connection with SmartZone using the SmartZone IP addresses provided via DHCP Option 43.

## Parameters

*ip-address*
The first SmartZone IP address that the switch will attempt to connect with.

*ip-address2*
SmartZone IP address that the switch will attempt to connect with if the first address doesn't work.

*ip-address3*
SmartZone IP address that the switch will attempt to connect with if the first two addresses don't work.

## Modes

Global configuration

## Usage Guidelines

Use the no form of the **sz active-list** command to remove the SmartZone IP addresses.

Beginning with SmartZone release 5.0, SmartZone can be used to monitor and manage ICX switches. An ICX switch identifies SmartZone and initiates a connection based on the SmartZone IP addresses configured on the switch with the **sz active-list** command or discovered through DHCP Option 43.

If the SmartZone IP addresses are both configured on the device and provided by DHCP Option 43, the configured list on the switch takes priority over DHCP Option 43. If none of the IP addresses in the configured list are reachable, the switch will try to reach the addresses received by DHCP Option 43.

Both "active" and "passive" SmartZone IP addresses can be configured on the ICX switch. Active IP addresses are given the highest priority; passive IP addresses are lowest priority and are provided for redundancy. The ICX switch will attempt to connect to SmartZone by sending a query to SmartZone IP addresses in the following order:

- SmartZone IP addresses configured on the ICX switch using the **sz active-list** command
- SmartZone IP addresses received through DHCP Option 43
- Backup SmartZone IP addresses configured on the ICX switch using the **sz passive-list** command

The switch connects to SmartZone using a reverse SSH tunnel.

# Examples

The following example shows the configuration of three active SmartZone IP addresses. The switch will attempt to create a connection beginning with the first IP address. If it fails, it will move to the second, and then third.

```
ICX(config)# sz active-list 192.168.11.200 192.168.11.201 192.168.11.202
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# sz disable

Disables SmartZone management of the device.

## Syntax

**sz disable**

**no sz disable**

## Command Default

SmartZone IP addresses are not configured on the switch, which means the switch will attempt to initiate a connection with SmartZone using the SmartZone IP addresses provided via DHCP Option 43.

## Modes

Global configuration

## Usage Guidelines

Use the no form of the **sz disable** command to reenable SmartZone management of the device after it has been disabled.

Beginning with SmartZone release 5.0, SmartZone can be used to monitor and manage ICX switches. An ICX switch identifies SmartZone and initiates a connection based on the SmartZone IP addresses configured on the switch or discovered through DHCP Option 43. When SmartZone management is disabled using the **sz disable** command, the ICX switch will not initiate a connection with SmartZone, even if SmartZone IP addresses are available.

Both "active" and "passive" SmartZone IP addresses can be configured on the ICX switch. Active IP addresses are given the highest priority; passive IP addresses are lowest priority and are provided for redundancy. The ICX switch will attempt to connect to SmartZone by sending a query to SmartZone IP addresses in the following order:

- SmartZone IP addresses configured on the ICX switch using the **sz active** command
- SmartZone IP addresses received through DHCP Option 43
- Backup SmartZone IP addresses configured on the ICX switch using the **sz passive** command

The switch connects to SmartZone using a reverse SSH tunnel.

## Examples

The following example shows SmartZone management disabled on the ICX device.

```
ICX(config)# sz disable
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# sz disconnect

Disconnects the switch from the current SmartZone connection and initiates a new connection to SmartZone based on the IP address list that is currently available.

## Syntax

**sz disconnect**

## Command Default

The switch has a connection with SmartZone.

## Modes

Privileged EXEC

## Usage Guidelines

Beginning with SmartZone release 5.0, SmartZone can be used to monitor and manage ICX switches. An ICX switch identifies SmartZone and initiates a connection based on the SmartZone IP addresses configured on the switch or discovered through DHCP Option 43.

Once the connection has been disconnected, the switch will initiate a new connection with SmartZone using the IP address list that is available and in the order that the IP address list is configured.

This command can be executed from both the local terminal as well as by SmartZone via Telnet over reverse-SSH.

## Examples

The following example shows the connection with SmartZone being disconnected.

```
ICX# sz disconnect
SZ Disconnect initiated…
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# sz passive

Configures the lowest priority SmartZone IP addresses that the ICX switch will use to initiate a connection with SmartZone.

## Syntax

**sz passive** { *ip-address* } [ *ip-address2* ] [ *ip-address3* ]

**no sz active**

## Command Default

SmartZone IP addresses are not configured on the switch, which means the switch will attempt to initiate a connection with SmartZone using the SmartZone IP addresses provided via DHCP Option 43.

## Parameters

*ip-address*
The first of the back-up SmartZone IP address that the switch will attempt to connect with.

*ip-address2*
SmartZone IP address that the switch will attempt to connect with if the first back-up address doesn't work.

*ip-address3*
SmartZone IP address that the switch will attempt to connect with if the first two back-up addresses don't work.

## Modes

Global configuration

## Usage Guidelines

Use the no form of the **sz passive** command to remove the SmartZone IP addresses.

An ICX switch identifies SmartZone and initiates a connection based on the SmartZone IP addresses configured on the switch or discovered through DHCP Option 43.

Both "active" and "passive" SmartZone IP addresses can be configured on the ICX switch. Active IP addresses are given the highest priority; passive IP addresses are lowest priority and are provided for redundancy. The ICX switch will attempt to connect to SmartZone by sending a query to SmartZone IP addresses in the following order:

- SmartZone IP addresses configured on the ICX switch using the **sz active** command
- SmartZone IP addresses received through DHCP Option 43
- Backup SmartZone IP addresses configured on the ICX switch using the **sz passive** command

If SmartZone IP addresses are configured on both the ICX switch and provided by DHCP Option 43, the configured list on the switch takes priority over DHCP Option 43. If none of the IP addresses in the configured list are reachable, the switch will try to reach the addresses received by DHCP Option 43.

The switch connects to SmartZone using a reverse SSH tunnel.

# Examples

The following example shows the configuration of three passive SmartZone IP addresses.

```
ICX(config)# sz passive 192.168.11.200 192.168.11.201 192.168.11.202
```

# History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# sz query

Initiates a query to a specific SmartZone IP address if it is not yet connected.

## Syntax

**sz query** *ip-address*

## Command Default

A query is not initiated.

## Modes

Privileged EXEC

## Usage Guidelines

Beginning with SmartZone release 5.0, SmartZone can be used to monitor and manage ICX switches. An ICX switch identifies SmartZone and initiates a connection based on the SmartZone IP addresses configured on the switch or discovered through DHCP Option 43.

## Examples

The following example shows a query sent to SmartZone.

```
ICX> sz query 104.198.196.69
Sending TA Request to IP: 104.198.196.69 Request POST URL: /switchm/api/v1/switch/auth/DUH3827L069
Response might take up to few mins...
Mar 29 22:41:32 Initiating HTTP POST Request...
Mar 29 22:41:38 HTTP Request Succeed for Request ID: 2. Response Code: 200
Received Buffer:
    <Received JSON Payload>
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was introduced. |

# sz registrar

Allows an administrator to override the default registrar host used in registrar-based SmartZone discovery.

## Syntax

**sz registrar** *hostname*

**no sz registrar** *hostname*

## Command Default

The default registrar host is sw-registrar.ruckuswireless.com.

## Parameters

*hostname*

Name of the registrar host in ASCII format.

## Modes

Global configuration mode

## Usage Guidelines

**NOTE**
This command is intended for testing purposes only.

The default registrar host is sw-registrar.ruckuswireless.com.

Use the **no** form of this command to disable registrar-based SmartZone discovery.

## Examples

The following example shows how to specify local-registrar.example.com as the registrar host.

```
device# configure terminal
device(config)# sz registrar local-registrar.example.com
```

To disable registrar-based SmartZone discovery, use the **no** form of the command as shown in the following example.

```
device# configure terminal
device(config)# no sz registrar sw-registrar.ruckuswireless.com
```

## History

| Release version | Command history |
|---|---|
| 08.0.80c | This command was introduced. |

# sz registrar-list

Allows administrators to clear addresses learned via the switch registrar from the SmartZone node list.

## Syntax

**no sz registrar-list** { *ip-address* } [ *ip-address* ]

## Parameters

*ip-address*

Specifies an IP addressed learned via the SmartZone registrar.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command clears the specified IP address from the SmartZone registrar list. Only the **no** form of the command is valid on ICX devices.

You can enter one or two IP addresses on the command line.

## Examples

The following example clears the entry for IP address 8.8.01 from the list of SmartZone IP addresses learned by the Ruckus switch registrar.

```
device(config)# no sz-registrar 8.8.0.1
```

## History

| Release version | Command history |
|---|---|
| 08.0.80c | This command was introduced. |

# sz registrar-query-restart

Allows administrators to force start the SmartZone discovery on ICX devices via the SmartZone registrar.

## Syntax

**sz registrar-query-restart**

## Modes

Global configuration mode

## Examples

The following example force starts the SmartZone discovery process on an ICX device.

```
device# configure terminal
device(config)# sz registrar-query-restart
```

## History

| Release version | Command history |
|---|---|
| 08.0.80c | This command was introduced. |

# table-map

Maps external entry attributes into the BGP routing table, ensuring that those attributes are preserved after being redistributed into OSPF.

## Syntax

**table-map** *string*

**no table-map** *string*

## Parameters

*string*

Specifies a route map to be whose attributes are to be preserved. Range is from 1 through 63 ASCII characters.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

BGP address-family IPv4 unicast VRF configuration mode

BGP address-family IPv6 unicast VRF configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

Use this command only to set the tag values. Normally, a route map is applied on routes (and therefore the routes are updated) before it is stored in the BGP routing table. Use the **table-map** command to begin the update before the routes are stored in the IP routing table.

Configurations made by this command apply to all peers.

Route maps that contain **set** statements change values in routes when the routes are accepted by the route map. For inbound route maps (route maps that filter routes received from neighbors), the routes are changed before they enter the BGP4 routing table. For tag values, if you do not want the value to change until a route enters the IP routing table, you can use a table map to change the value. A table map is a route map that you have associated with the IP routing table. The device applies the **set** statements for tag values in the table map to routes before adding them to the routing table. To configure a table map, you first configure the route map, then identify it as a table map. The table map does not require separate configuration. You can have only one table map.

**NOTE**
Use table maps only for setting the tag value. Do not use table maps to set other attributes. To set other route attributes, use route maps or filters. To create a route map and identify it as a table map, enter commands such those shown in the first example below. These commands create a route map that uses an address filter. For routes that match the IP prefix list filter, the route map changes the tag value to 100 and is then considered as a table map. This route map is applied only to routes that the device places in the IP routing table. The route map is not applied to all routes. The first example below assumes that IP prefix list p11 has already been configured.

The **no** form of the command removes the table map.

# Examples

The following example illustrates the execution of the **table-map** command.

```
device# configure terminal
device(config)# route-map tag_ip permit 1
device(config-route-map tag_ip)# match ip address prefix-list p11
device(config-route-map tag_ip)# set tag 100
device(config-route-map tag_ip)# exit
device(config)# router bgp
device(config-bgp-router)# table-map tag_ip
```

The following example removes a table map in the IPv6 address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# no table-map tag_ip
```

# tacacs-server deadtime

Configures the duration for which the device waits for the primary authentication server to reply before deciding the TACACS server is dead and trying to authenticate using the next server.

## Syntax

**tacacs-server deadtime** *time*

**no tacacs-server deadtime** *time*

## Command Default

The default duration is three seconds.

## Parameters

*time*

The time in seconds. The valid values are from 1 through 5. The default is 3.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the duration for which the device waits for a reply before deciding that the server is dead.

## Examples

The following example configures the dead time as four seconds.

```
device(config)# tacacs-server deadtime 4
```

# tacacs-server enable

Configures the device to allow TACACS server management access only to clients connected to ports within port-based VLAN.

## Syntax

**tacacs-server enable vlan** *vlan-number*

**no tacacs-server enable vlan** *vlan-number*

## Command Default

By default, access is allowed on all ports.

## Parameters

**vlan** *vlan-number*

Configures access only to clients connected to ports within the VLAN.

## Modes

Global configuration mode

## Usage Guidelines

You can restrict management access to a device to ports within a specific port-based VLAN. VLAN-based access control works in conjunction with other access control methods. Clients connected to ports that are not in the VLAN are denied management access.

As in a switched network, the TACACS server and the SSH client should be included in the same VLAN. Otherwise, the response expected from the TACACS server should be sent in the same VLAN configured by the **tacacs-server enable vlan** command. This configuration allows the TACACS server to be in a different VLAN and still allow SSH connections in a routed network.

The **tacacs-server enable vlan** command should not be configured in a network that uses dynamic routing, where the TACACS server response might be routed on any path.

The **no** form of the command removes the restriction.

## Examples

The following example shows how to allow TACACS server access only to clients in a specific VLAN.

```
device(config)# tacacs-server enable vlan 10
```

# tacacs-server host

Configures the TACACS server host to authenticate access to a device.

## Syntax

**tacacs-server host** { *ipv4-address* | *host-name* | *ipv6-address*} [ **auth-port** *port-num* [ **acct-port** *port-num* [ { **accounting-only** | **authentication-only** | **authorization-only** | **default** } [ **key** *key-string* ] ] ]

**no tacacs-server host** { *ipv4-address* | *host-name* | *ipv6-address*} [ **auth-port** *port-num* [ **acct-port** *port-num* [ { **accounting-only** | **authentication-only** | **authorization-only** | **default** } [ **key** *key-string* ] ] ]

## Command Default

The TACACS server host is not configured.

## Parameters

*ipv4-address*
Configures the IPv4 address of the TACACS server.

*host-name*
Configures the host name of the TACACS server.

*ipv6-address*
Configures the IPv6 address of the TACACS server.

**auth-port** *port-num*
Configures the authentication UDP port. The default value is 1812.

**acct-port** *port-num*
Configures the accounting UDP port. The default value is 1813.

**accounting-only**
Configures the server to be used only for accounting. Supported for TACACS+ only.

**authentication-only**
Configures the server to be used only for authentication. Supported for TACACS+ only.

**authorization-only**
Configures the server to be used only for authorization. Supported for TACACS+ only.

**default**
Configures the server to be used for any AAA operation. Supported for TACACS+ only.

**key** *key-string*
Configures the TACACS key for the server. Supported for TACACS+ only.

## Modes

Global configuration mode

# Usage Guidelines

You can specify up to eight servers. If you add multiple TACACS or TACACS+ authentication servers to the device, the device tries to reach them in the order you add them. To use a TACACS server to authenticate access to a device, you must identify the server to the device. In a TACACS configuration, you can designate a server to handle a specific AAA task. For example, you can designate one TACACS server to handle authorization and another TACACS server to handle accounting. You can specify individual servers for authentication and accounting, and authorization. You can set the TACACS key for each server.

The **no** form of this command removes the configuration.

# Examples

The following example shows how to configure a TACACS server to authenticate access to a device.

```
device(config)# tacacs-server host 192.168.10.1
```

The following example shows how to specify different TACACS servers for authentication and accounting.

```
device(config)# tacacs-server host 10.2.3.4 auth-port 1800 acct-port 1850 default key abc
device(config)# tacacs-server host 10.2.3.5 auth-port 1800 acct-port 1850 authentication-only key def
device(config)# tacacs-server host 10.2.3.6 auth-port 1800 acct-port 1850 accounting-only key ghi
```

# tacacs-server key

Configures the value that the device sends to the TACACS server when trying to authenticate user access.

## Syntax

**tacacs-server key** *key-string*

**no tacacs-server key** *key-string*

## Command Default

The TACACS server key is not configured.

## Parameters

*key-string*

Specifies the key as an ASCII string. The value for the key parameter on the device should match the one configured on the TACACS server. The key can be from 1 to 32 characters in length and cannot include any space characters.

## Modes

Global configuration mode

## Usage Guidelines

The **tacacs-server key** command is used to encrypt TACACS packets before they are sent over the network.

The **no** form of the command removes the TACACS server key configuration.

## Examples

The following example shows how to configure a TACACS server key.

```
device(config)# racacs-server key abc
```

# tacacs-server retransmit

Configures the maximum number of retransmission attempts for a request when a TACACS authentication request times out.

## Syntax

**tacacs-server retransmit** *number*

**no tacacs-server retransmit** *number*

## Command Default

The default number of retries is three.

## Parameters

*number*

The maximum number of retries that the software retransmits the request. The valid values are from 1 through 5. The default is 3.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command resets the maximum number of retransmission attempts to the default.

## Examples

The following example shows how to set the maximum number of retransmission attempts to four.

```
device(config)# tacacs-server retransmission 4
```

# tacacs-server timeout

Configures the number of seconds the device waits for a response from a TACACS server before either retrying the authentication request or determining that the TACACS servers are unavailable and moving on to the next authentication method in the authentication method list.

## Syntax

**tacacs-server timeout** *time*

**no tacacs-server timeout** *time*

## Command Default

The default timeout value is three seconds.

## Parameters

*time*

The time in seconds. Valid values are from 1 through 15 seconds. The default is 3.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the timeout value to the default.

## Examples

The following example shows how to set the TACACS server timeout value to 10 seconds.

```
device(config)# tacacs-server timeout 10
```

# tag-profile

Configures or changes the tag profile for 802.1ad tagging.

## Syntax

**tag-profile** *tag-number*

**no tag-profile** *tag-number*

## Command Default

The default tag number is 0x8100.

## Parameters

*tag-number*

Specifies the number of the tag. The value can be 0x8100 (default) or 0xffff.

## Modes

Global configuration mode

## Usage Guidelines

Tag profiles on a single port, or a group of ports, can be configured to point to the global tag profile.

The **no** command removes the tag profile configuration.

## Examples

The following example shows how to configure the tag profile.

```
device(config)# tag-profile 9500
```

# tag-profile enable

Directs the individual ports or a range of ports to the tag profile.

## Syntax

**tag-profile enable**

**no tag-profile enable**

## Command Default

The tag profile is not enabled.

## Modes

Interface configuration mode

## Usage Guidelines

Tag profiles on a single port, or a group of ports, can be configured to point to the global tag profile.

The tag type and tag profile cannot be configured at the same time.

The **no** form of the command disables the tag profile for ports.

## Examples

The following example shows how to enable tag profile for a single port.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# tag-profile enable
```

The following example shows how to enable tag profile for multiple ports.

```
device(config)# interface ethernet 1/1/1 ethernet 1/2/1
device(config-mif-1/1/1,1/2/1)# tag-profile enable
```

# tagged ethernet

Tags a port to allow communication among the different VLANs to which the port is assigned.

## Syntax

**tagged ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ]

**no tagged ethernet** *stackid/slot/port* [ **to** *stackid/slot/port* | [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ] [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ]... ]

## Parameters

**ethernet** *stackid/slot/port*

Specifies the Ethernet interface to configure as a tagged port.

**to** *stackid/slot/port*

Specifies a range of Ethernet interfaces.

## Modes

VLAN configuration mode

## Usage Guidelines

Tagging does not apply to the default VLAN. The ports are defined as either tagged or untagged at the VLAN level.

The **no** form of the command removes the tagging of the Ethernet ports.

## Examples

The following example tags the port 1/1/9 to VLAN 4.

```
device# configure terminal
device(config)# vlan 4
device(config-vlan-4)# tagged ethernet 1/1/9
```

The following example tags the port 1/12 to a RSPAN (Remote Switched Port Analyzer) VLAN.

```
device# configure terminal
device(config)# rspan-vlan 4000
device(config-rspan-vlan)# tagged ethernet 1/1/2
device(config-rspan-vlan)# rspan destination ethernet 1/1/2
```

## History

| Release version | Command history |
|---|---|
| 08.0.80 | This command was modified to add support for RSPAN configuration mode. |

# tagged lag

Tags LAG virtual interfaces to allow communication among the different VLANs to which the LAG virtual interface is assigned.

## Syntax

**tagged lag** *lag-id* [ **to** *lag-id* | [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ] [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ]

**no tagged lag** *lag-id* [ **to** *lag-id* | [ **lag** *lag-id* **to** *lag-id* | **lag** *lag-id* ] [ **ethernet** *stackid/slot/port* **to** *stackid/slot/port* | **ethernet** *stackid/slot/port* ]... ]

## Parameters

**lag** *lag-id*

Specifies the LAG virtual interface.

**to** *lag-id*

Specifies a range of LAG IDs.

**ethernet** *stackid/slot/port*

Specifies the Ethernet interface to configure as a tagged port.

**to** *stackid/slot/port*

Specifies a range of Ethernet interfaces.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command removes the tagging of the LAG virtual interfaces.

## Examples

The following example tags the LAG virtual interfaces 1 to 3 to VLAN 4.

```
device# configure terminal
device(config)# vlan 4
device(config-vlan-4)# tagged lag 1 to 3
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# telnet

Enables a Telnet connection from the device to a remote IPv6 host using the console.

## Syntax

**telnet** { *host-name* | *host-ipaddress* } [ *remote-port-num* ]

**telnet** { *host-name* | *host-ipv6address* } [ **outgoing-interface** { **ethernet** *stack/slot/port* | **ve** *ve-num* } ] [ *remote-port-num* ]

## Parameters

*host-name*
Specifies the host name of the remote host.

*host-ipaddress*
Specifies the IPv4 address of the remote host.

*remote-port-num*
Specifies the port number on which the device establishes theTelnet connection. Valid values are 1 to 65535. If you do not specify a port number, the device establishes the Telnet connection on port 23.

*host-ipv6address*
Specifies the IPv6 address of the remote host.

**outgoing-interface**
Identifies the interface that must be used to reach the remote host.

**ethernet** *stack/slot/port*
Identifies the Ethernet interface that must be used to reach the remote host.

**ve** *ve-num*
Identifies the VE interface that must be used to reach the remote host.

## Modes

Privileged EXEC mode

## Usage Guidelines

The **telnet** command establishes a Telnet connection from a device to a remote host using the console. Up to five read-access Telnet sessions are supported on the router at one time. Write-access through Telnet is limited to one session, and only one outgoing Telnet session is supported on the router at one time. To see the number of open Telnet sessions at any time, enter the **show telnet** command.

## Examples

The following example establishes a Telnet connection to a remote host with the IPv6 address of 2001:DB8:3de2:c37::6.

```
device# telnet 2001:DB8:3de2:c37::6
```

# telnet access-group

Configures an ACL that restricts Telnet access to the device.

## Syntax

**telnet access-group** { *acl-num* | *acl-name* | **ipv6** *ipv6-acl-name* }

**no telnet access-group** { *acl-num* | *acl-name* | **ipv6** *ipv6-acl-name* }

## Command Default

Telnet access is not restricted.

## Parameters

*acl-num*
The standard access list number. The valid values are from 1 through 99.

*acl-name*
The standard access list name.

**ipv6** *ipv6-acl-name*
The IPv6 access list name.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the Telnet access restriction.

## Examples

The following example shows how to configure an ACL that restricts Telnet access to the device. In this example, ACL 10 is configured. The device allows Telnet access to all IP addresses except those listed in ACL 10.

```
device(config)# access-list 10 deny host 10.157.22.32 log
device(config)# access-list 10 deny 10.157.23.0 0.0.0.255 log
device(config)# access-list 10 deny 10.157.24.0 0.0.0.255 log
device(config)# access-list 10 deny 10.157.25.0/24 log
device(config)# access-list 10 permit any
device(config)# telnet access-group 10
device(config)# write memory
```

# telnet client

Restricts Telnet access to a host with the specified IP address.

## Syntax

**telnet client** { *ipv4-address* [ *client-mac* ] | **any** *client-mac* | **ipv6** *ipv6-address*}

**no telnet client** { *ipv4-address* [ *client-mac* ] | **any** *client-mac* | **ipv6** *ipv6-address*}

## Command Default

Remote Telnet access is not restricted.

## Parameters

*ipv4-address*
: Allows Telnet access only to the host with the IPv4 address.

*client-mac*
: The host MAC address.

**any** *client-mac*
: Allows Telnet access to a host with any IP address but with the specified MAC address.

**ipv6** *ipv6-address*
: Allows Telnet access to a host with the specified IPv6 address.

## Modes

Global configuration mode

## Usage Guidelines

You can specify only one IP address with one command. However, you can enter the command ten times to specify up to ten IP addresses.

The **no** form of the command removes the restriction and allows Telnet access to all the clients.

## Examples

The following example shows how to allow Telnet access only to the host with IP address 192.168.10.1 and MAC address 1111.2222.3333.

```
device(config)# telnet client 192.168.10.1 1111.2222.3333
```

# telnet login-retries

Configures the number of attempts you can enter a correct username and password before the device disconnects the Telnet session.

## Syntax

**telnet login-retries** *number*

**no telnet login-retries** *number*

## Command Default

By default, four attempts are supported.

## Parameters

*number*

The number of retries the device prompts you for a username and password before disconnecting the Telnet session. The valid values are from 0 through 5. The default is 4.

## Modes

Global configuration mode

## Usage Guidelines

If you are connecting to the device using Telnet, the device prompts you for a username and password. By default, you have up to four chances to enter a correct username and password. If you do not enter a correct username or password after four attempts, the device disconnects the Telnet session.

The **no** form of the command resets the number of attempts to the default.

**NOTE**
You must configure Telnet with the **enable telnet authentication local** command to enable only a specific number of Telnet login attempts.

## Examples

The following example shows how to configure up to five chances to enter a correct username and password before getting disconnected.

```
device(config)# telnet login-retries 5
```

# telnet login-timeout

Configures the login timeout for a Telnet session.

## Syntax

**telnet login-timeout** *time*

**no telnet login-timeout** *time*

## Command Default

The default login timeout is one minute.

## Parameters

*time*

Time in minutes. The valid values are from 1 through 10. The default is 1.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command sets the login timeout value to the default.

## Examples

The following example shows how to set the login timeout value of a Telnet session to ten minutes.

```
device(config)# telnet login-timeout 10
```

# telnet server enable

Configures Telnet access only to clients in a specific VLAN.

## Syntax

**telnet server enable vlan** *vlan-num*

**no telnet server enable vlan** *vlan-num*

## Command Default

Telnet access is not restricted.

## Parameters

**vlan** *vlan-num*

Configures access only to clients connected to ports within the VLAN.

## Modes

Global configuration mode

## Usage Guidelines

You can restrict Telnet access to a Ruckus device to ports within a specific port-based VLAN. VLAN-based access control works in conjunction with other access control methods. Clients connected to ports that are not in the VLAN are denied management access.

The **no** form of the command allows Telnet access to all clients.

## Examples

The following example shows how to allow Telnet access only to clients connected to ports within port-based VLAN 40.

```
device(config)# telnet server enable vlan 40
```

# telnet server suppress-reject-message

Configures the device to suppress the Telnet connection rejection message.

## Syntax

**telnet server suppress-reject-message**

**no telnet server suppress-reject-message**

## Command Default

Rejection messages are sent.

## Modes

Global configuration mode

## Usage Guidelines

By default, if a device denies Telnet management access to the device, the software sends a message to the denied Telnet client. You can optionally suppress the rejection message. When you enable the option, a denied Telnet client does not receive a message from the device. Instead, the denied client simply does not gain access.

The **no** form of the command configures the device to send the rejection message.

## Examples

The following example shows the configuration to suppress the connection rejection message sent by the device to a denied Telnet client.

```
device(config)# telnet server suppress-reject-message
```

# telnet strict-management-vrf

Allows incoming Telnet connection requests only from the management VRF and not from the out-of-band (OOB) management port.

## Syntax

**telnet strict-management-vrf**

**no telnet strict-management-vrf**

## Command Default

When the management VRF is configured, incoming Telnet connection requests are allowed from the ports that belong to the management VRF and from the OOB management port.

## Modes

Global configuration mode

## Usage Guidelines

The **telnet strict-management-vrf** command is applicable only when the management VRF is configured. If a management VRF is not configured, configuring the **telnet strict-management-vrf** command displays an error message.

The **telnet strict-management-vrf** command does not prevent a connection initiated from the OOB management interface if the management interface VRF and the management VRF are the same. The user must configure either the **management exclude all oob** command or the **management exclude telnet oob** command.

For the Telnet server, changing the management VRF configuration or configuring the **telnet strict-management-vrf** command does not affect the existing Telnet connections. The changes are applied only to new incoming connection requests.

The **telnet strict-management-vrf** command and the **management exclude** commands are mutually exclusive. If the latter command is configured, outbound Telnet connections are not blocked.

The **no** form of the command enables the incoming Telnet connection requests from ports that belong to the management VRF and from the out-of-band management port.

## Examples

The following example allows incoming Telnet connection requests from the management VRF only.

```
device(config)# telnet strict-management-vrf
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# Related Commands

management exclude

# telnet timeout

Configures the duration of time, a Telnet session can remain idle before it is timed out.

## Syntax

**telnet timeout** *time*

**no telnet timeout** *time*

## Command Default

The Telnet session never times out.

## Parameters

*time*

The time in minutes. The valid values are from 0 through 240. The default is 0; the session never times out.

## Modes

Global configuration mode

## Usage Guidelines

An idle Telnet session is a session that is still sending TCP ACKs in response to keep alive messages from the device, but is not being used to send data.

The **no** form of the command resets the default timeout value.

## Examples

The following example shows how to set the Telnet session idle timeout to 100 minutes.

```
device(config)# telnet timeout 100
```

# temperature warning

Changes the temperature threshold at which the device sends a syslog message and an SNMP trap.

## Syntax

**temperature warning** { *stack-id temp-threshold* }

## Command Default

The default threshold varies by the hardware device. Refer to the hardware installation guide for your device.

## Parameters

*stack-id*

Stack number. Value is from 1 to 8.

*temp-threshold*

Temperature warning level, in Celsius. See Usage Guidelines for more details.

## Modes

Privileged EXEC mode

## Usage Guidelines

There is no **no** form of this command.

When setting the *temp-threshold* option, you must not set this level higher than the maximum value allowed by your device. The temperature warning level must be at least five degrees Celsius less than the temperature shutdown level, which is automatically set by the device.

## Examples

The following example sets the temperature threshold to 75°C.

```
device# temperature warning 1 75
```

# terminal logging

Disables or re-enables terminal logging, which captures all the console prints generated on the system to a RAMFS file and copies the RAMFS file to the flash memory upon certain triggers.

## Syntax

**terminal logging**

**no terminal logging**

## Command Default

Terminal logging is enabled by default.

## Modes

Global configuration mode

## Usage Guidelines

The file size is limited to 10 MB after which the prints wrap over.

Console prints are stored in the ss_console.txt file. Terminal logging also logs dmesg output (Linux kernel log) in the kmsg.txt file and copies it to flash memory.

The terminal logging files are stored in the /fast_iron/logs folder.

The log files copied to the flash memory can be retrieved later using supportsave for offline debugging and analysis.

Logs from Telnet and SSH sessions are also logged to the file.

The **no** form of the command disables terminal logging.

## Examples

The following example disables terminal logging.

```
device# configure terminal
device(config)# no terminal logging
Terminal Logging Feature is now disabled
```

The following example re-enables terminal logging.

```
device# configure terminal
device(config)# terminal logging
Terminal Logging Feature is now enabled
```

## History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was introduced. |

# terminal monitor

Enables the real-time display for a Telnet or SSH session.

## Syntax

**terminal monitor**

## Command Default

Real-time display is not enabled.

## Modes

Privileged EXEC mode

## Usage Guidelines

The command toggles the feature on and off. The CLI displays a message to indicate the status change for the feature. To enable or disable the feature in the management session, enter the **terminal monitor** command again.

Any terminal logged on to a Ruckus switch can receive real-time Syslog messages when the **terminal monitor** command is issued.

## Examples

The following example enables real-time display for a Telnet or SSH session.

```
device# terminal monitor
Syslog trace was turned ON
SYSLOG: <9>device, Power supply 2, power supply on left connector, failed
SYSLOG: <14>device, Interface ethernet 6, state down
SYSLOG: <14>device, Interface ethernet 2, state up
```

The following example disables real-time display for a Telnet or SSH session.

```
device# terminal monitor
Syslog trace was turned OFF
```

# tftp client enable

Configures the device to allow TFTP access only to clients in a specific VLAN.

## Syntax

**tftp client enable vlan** *vlan-num*

**no tftp client enable vlan** *vlan-num*

## Command Default

TFTP client access is enabled for all the clients.

## Parameters

**vlan** *vlan-num*

Configures access only to clients connected to ports within the VLAN.

## Modes

Global configuration mode

## Usage Guidelines

You can restrict TFTP access to a Ruckus device to ports within a specific port-based VLAN. VLAN-based access control works in conjunction with other access control methods. Clients connected to ports that are not in the VLAN are denied management access.

The **no** form of the command allows access to all clients.

## Examples

The following example shows how to allow TFTP access only to clients connected to ports within port-based VLAN 40.

```
device(config)# tftp client enable vlan 40
```

# tftp disable

Disables TFTP client access.

## Syntax

**tftp disable**

**no tftp disable**

## Command Default

TFTP client access is enabled.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command enables TFTP client access.

## Examples

The following example shows how to disable TFTP client access.

```
device(config)# tftp disable
```

# tftp-server

Specifies the address or name of the TFTP server to be used by the DHCP client.

## Syntax

**tftp-server** { **address** | **name** *server-name* }

## Parameters

**address**

Specifies the IP address of the DHCP server.

**name** *server-name*

Configures the TFTP server specified by the server name.

## Modes

DHCP server pool configuration mode.

## Usage Guidelines

If DHCP options 66 (TFTP server name) and option 150 (TFTP server IP address) are both configured, the DHCP client ignores option 150 and tries to resolve the TFTP server name (option 66) using DNS.

## Examples

The following example specifies the TFTP server to be used by the DHCP client.

```
device(config)# ip dhcp-server-pool cabo
device(config-dhcp-cabo)# tftp-server 10.7.5.48
```

# tftp-server (IMAGE)

Configures the TFTP server location where auto image copy can download a software image.

## Syntax

**tftp-server** *ip-address* **image-location** *path*

**no tftp-server** *ip-address* **image-location** *path*

## Command Default

No TFTP server location is configured for auto image copy downloads.

## Parameters

*ip-address*
Specifies the IP address of the TFTP server.

**image-location** *path*
Specifies the directory path to the software image on the TFTP server.

## Modes

Global configuration mode

## Usage Guidelines

To avoid image mismatch issues, set up a TFTP server for auto image copy before configuring a stack.

The **no** form of the command removes the TFTP server configuration.

## Examples

The following example specifies a TFTP server location where a software image is located:

```
device(config)# tftp-server 10.1.2.1 image-location /server/builds/
```

## History

| Release version | Command history |
|---|---|
| 8.0.00a | This command was introduced. |

# timers (BGP)

Adjusts the interval at which BGP KEEPALIVE and HOLDTIME messages are sent.

## Syntax

**timers** { **keep-alive** *keepalive_interval* **hold-time** *holdtime_interval* }

**no timers**

## Parameters

**keep-alive** *keepalive_interval*

Frequency in seconds with which a device sends keepalive messages to a peer. Range is from 0 through 65535 seconds. The default is 60.

**hold-time** *holdtime_interval*

Interval in seconds that a device waits to receive a keepalive message from a peer before declaring that peer dead. Range is from 0 through 65535 seconds. The default is 180.

## Modes

BGP configuration mode

## Usage Guidelines

The KEEPALIVE and HOLDTIME message interval is overwritten when the **fast-external-failover** command takes effect on a down link to a peer.

You must enter a value for **keep-alive** before you can enter a value for **hold-time**. Both values must be entered. If you only want to adjust the value of one parameter, enter the default value of the parameter that you do not want to adjust.

The **no** form of the command clears the timers.

## Examples

The following example sets the keepalive timer for a device to 120 seconds and the hold-timer to 360 seconds.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# timers keep-alive 120 hold-time 360
```

# timers (OSPFv2)

Configures Link State Advertisement (LSA) pacing and Shortest Path First (SPF) throttle timers.

## Syntax

**timers** { **lsa-group-pacing** *interval* | **throttle spf** *start hold max* }

## Command Default

See the parameters section for specific defaults.

## Parameters

**lsa-group-pacing** *interval*

Specifies the interval at which OSPF LSAs are collected into a group and refreshed, check-summed, or aged by the OSPF process. Valid values range from 10 to 1800 seconds. The default is 240 seconds.

**throttle spf**

Specifies start, hold and maximum wait intervals for throttling SPF calculations for performance. The values you enter are in milliseconds.

*start*

Initial SPF calculation delay. Valid values range from 0 to 60000 milliseconds. The default is 0.

*hold*

Minimum hold time between two consecutive SPF calculations. Valid values range from 0 to 60000 milliseconds. The default is 0.

*max*

Maximum wait time between two consecutive SPF calculations. Valid values range from 0 to 60000 milliseconds. The default is 0.

## Modes

OSPF router configuration mode

OSPF VRF router configuration mode

## Usage Guidelines

The device paces LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA refresh timer expires. The accumulated LSAs constitute a group, which the device refreshes and sends out together in one or more packets.

The LSA pacing interval is inversely proportional to the number of LSAs the device is refreshing and aging. For example, if you have a large database of 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a small database of about 100 LSAs, increasing the pacing interval to 10 to 20 minutes may enhance performance.

The **no timers lsa-group-pacing** command restores the pacing interval to its default value.

The **no timers throttle spf** command sets the SPF timers back to their defaults.

# Examples

The following example sets the LSA group pacing interval to 30 seconds.

```
device# configure terminal
device(config)# router ospf
device(config-ospf router)# timers lsa-group-pacing 30
```

The following example sets the SPF delay to 10000 milliseconds, the hold time to 15000 milliseconds, and the maximum wait time to 30000 milliseconds.

```
device# configure terminal
device(config)# router ospf
device(config-ospf router)# timers throttle spf 10000 15000 30000
```

# timers (OSPFv3)

Configures Link State Advertisement (LSA) pacing and Shortest Path First (SPF) timers.

## Syntax

**timers** { **lsa-group-pacing** *interval* | **spf** *start hold* }

## Command Default

Enabled.

## Parameters

**lsa-group-pacing** *interval*

Specifies the interval at which OSPFv3 LSAs are collected into a group and refreshed, check-summed, or aged by the OSPFv3 process. Valid values range from 10 to 1800 seconds. The default is 240 seconds.

**spf**

Specifies start and hold intervals for SPF calculations for performance. The values you enter are in milliseconds.

*start*

Initial SPF calculation delay. Valid values range from 0 to 65535 seconds. The default is 5 seconds.

*hold*

Minimum hold time between two consecutive SPF calculations. Valid values range from 0 to 65535 seconds. The default is 10 milliseconds.

## Modes

OSPFv3 router configuration mode

OSPFv3 router VRF configuration mode

## Usage Guidelines

The device paces LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA refresh timer expires. The accumulated LSAs constitute a group, which the device refreshes and sends out together in one or more packets.

The LSA pacing interval is inversely proportional to the number of LSAs the device is refreshing and aging. For example, if you have a large database of 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a small database of about 100 LSAs, increasing the pacing interval to 10 to 20 minutes may enhance performance.

The **no timers lsa-group-pacing** command restores the pacing interval to its default value.

The **no timers spf** command sets the SPF timers back to their defaults.

# Examples

The following example sets the LSA group pacing interval to 30 seconds.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# timers lsa-group-pacing 30
```

The following example sets the SPF delay time to 10 and the hold time to 20.

```
device# configure terminal
device(config)# ipv6 router ospf
device(config-ospf6-router)# timers spf 10 20
```

# timers (RIP)

Specifies how often RIP update messages are sent.

## Syntax

**timers** { *update-timer time-out-timer hold-down-timer garbage-collection-timer* }

**no timers** { *update-timer time-out-timer hold-down-timer garbage-collection-timer* }

## Command Default

Defaults differ by timer. Refer to timer parameter descriptions.

## Parameters

*update-timer*

Sets the amount of time between RIP routing updates. The default is 30 seconds. Possible values are 3 through 21845 seconds.

*timeout-timer*

Sets the amount of time after which a route is considered unreachable. The default is 180 seconds. Possible values are 9 through 65535 seconds.

*hold-down-timer*

Sets the amount of time during which information about other paths is ignored. The default is 180 seconds. Possible values are 0 through 65535 seconds.

*garbage-collection-timer*

Sets the amount of time after which a route is removed from the RIP routing table. The default is 120 seconds. Possible values are 0 through 65535.

## Modes

RIP router configuration mode.

## Usage Guidelines

The **no** form of the command returns all timers to their default settings.

RIP must be enabled before you can set the timers. All timer values, including values that are not being modified, must be present when you enter the command.

## Examples

The following command sets the RIP update timer to 30 seconds, the RIP timeout timer to 180 seconds, the RIP hold-down timer to 185 seconds, and the RIP garbage collection timer to 120 seconds.

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# timer 30 180 185 120
```

# timers (RIPng)

Adjusts RIPng timers.

## Syntax

**timers** { *update-timer time-out-timer hold-down-timer garbage-collection-timer* }

**no timers** { *update-timer time-out-timer hold-down-timer garbage-collection-timer* }

## Command Default

Defaults differ by timer. Refer to timer parameter descriptions.

## Parameters

*update-timer*

Sets the amount of time between RIPng routing updates. The default is 30 seconds. Possible values are 3 through 65535 seconds.

*timeout-timer*

Sets the amount of time after which a route is considered unreachable. The default is 180 seconds. Possible values are 9 through 65535 seconds.

*hold-down-timer*

Sets the amount of time during which information about other paths is ignored. The default is 180 seconds. Possible values are 9 through 65535 seconds.

*garbage-collection-timer*

Sets the amount of time after which a route is removed from the RIPng routing table. The default is 120 seconds. Possible values are 9 through 65535.

## Modes

RIPng router configuration mode

## Usage Guidelines

The **no** form of the command returns the timers to their default settings.

RIPng must be enabled before you can set the timers.

You must enter values for all of the timers, even those you do not want to reset. This is true for the **no** form of the command as well.

# Examples

The following example adjusts the setting for the garbage collection timer and retains default settings for all other timers.

```
device# configure terminal
device(config)# ipv6 router rip
device(config-ripng-router)# timers 30 180 180 110
```

# timeout (EFM-OAM)

Configures the time in seconds for which the local Data Terminal Equipment (DTE) waits to receive OAM Protocol Data Units (OAMPDUs) from the remote entity.

## Syntax

**timeout** *value*

**no timeout** *value*

## Command Default

The default value is 5 seconds.

## Parameters

*value*

Specifies the time in seconds for which the local DTE must wait for OAMPDUs from the remote entity. The value range can be from 1 through 10 seconds.

## Modes

EFM-OAM protocol configuration mode

## Usage Guidelines

If the local DTE does not receive any OAMPDU within the specified period, the peer is considered down and the EFM-OAM discovery process will start over again.

The **no** form of the command restores the default value of 5 seconds.

## Examples

The following example configures the timeout value as 10 seconds.

```
device(config)# link-oam
device(config-link-oam)# timeout 10
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# tolerance

Configures the tolerance value for the accept keys and send keys for the keychain to extend the lifetime of the keys beyond the active lifetime duration (prior to the start of the lifetime or after the end of the lifetime).

## Syntax

**tolerance** *value*

**no tolerance** *value*

## Parameters

*value*

Specifies the tolerance duration in seconds. The valid range is from 1 through 8640000 seconds.

## Modes

Keychain configuration mode

## Usage Guidelines

If the tolerance value is configured, the start time of the key to become active is advanced (start time minus tolerance) and the end time is moved further ahead (end time plus tolerance) before the key expires, unless the end time is set to be infinite.

A key is considered valid when it is in the tolerance period.

The **no** form of the command removes the tolerance value added to all the keys under the keychain.

## Examples

The following example configures the keychain with a tolerance value of 600 seconds (10 minutes).

```
device# configure terminal
device(config)# keychain xprotocol
device(config-keychain-xprotocol)# tolerance 600
```

## History

| Release | Command History |
|---|---|
| 08.0.70 | This command was introduced. |

# topology-group

Configures the topology group.

## Syntax

**topology-group** *group-id*

**no topology-group** *group-id*

## Command Default

A topology group is not configured.

## Parameters

*group-id*

Specifies the topology group ID. The ID ranges from 1 through 256.

## Modes

Global configuration mode

## Usage Guidelines

Each topology group contains a master VLAN and can contain one or more member VLANs and VLAN groups. You must configure the master VLAN and member VLANs or member VLAN groups before you configure the topology group.

You can configure up to 30 topology groups. Each group can control up to 4096 VLANs. A VLAN cannot be controlled by more than one topology group. The topology group must contain a master VLAN and can also contain individual member VLANs, VLAN groups, or a combination of individual member VLANs and VLAN groups.

The **no** form of the command removes the topology group.

## Examples

The following example configures the topology group with ID 2 and adds master VLAN and member VLANs.

```
device# configure terminal
device(config)# topology-group 2
device(config-topo-group-2)# master-vlan 2
device(config-topo-group-2)# member-vlan 3
device(config-topo-group-2)# member-vlan 4
device(config-topo-group-2)# member-vlan 5
```

# traceroute

Determines the path through which a Ruckus device can reach another device.

## Syntax

**traceroute** [ **vrf** *vrf-name* ] *ipv4-address* [ **source-ip** *ip-address* ] [ **minttl** *min-value* ] [ **maxttl** *max-value* ] [ **numeric** ] [ **timeout** *value* ]

**traceroute** *host-name* [ **source-ip** *ip-address* ] [ **minttl** *min-value* ] [ **maxttl** *max-value* ] [ **numeric** ] [ **timeout** *value* ]

**traceroute ipv6** [ **vrf** *vrf-name* ] *ipv6-address* [ **minttl** *min-value* ] [ **maxttl** *max-value* ] [ **numeric** ] [ **timeout** *value* ]

**traceroute ipv6** *host-name* [ **minttl** *min-value* ] [ **maxttl** *max-value* ] [ **numeric** ] [ **timeout** *value* ]

## Parameters

**vrf** *vrf-name*
Specifies the Virtual Routing and Forwarding (VRF) instance.

*ipv4-address*
Specifies the host IPv4 address.

**source-ip** *ip-address*
Configures an IP address to be used as the origin for the traceroute.

**minttl** *min-value*
Specifies the minimum Time to Live (TTL) value (hops). The value can range from 1 through 255. The default value is 1.

**maxttl** *max-value*
Specifies the maximum TTL value (hops). The value can range from 1 through 255. The default value is 30.

**numeric**
Displays IP addresses in number format instead by name.

**timeout** *value*
Configures echo request timeout, in seconds. The value can range from 1 through 120. The default value is 2.

*host-name*
Specifies the host name.

**ipv6**
Displays IPv6-related information.

*ipv6-address*
Specifies the host IPv6 address.

## Modes

User EXEC mode

Privileged EXEC mode

## Usage Guidelines

The CLI displays trace-route information for each hop as soon as the information is received. Traceroute requests display all responses to a given TTL. In addition, if there are multiple equal-cost routes to the destination, the device displays up to three responses by default.

## Examples

The following example issues an IPv4 traceroute.

```
device> traceroute 10.33.4.7
```

The following example issues an IPv6 traceroute.

```
device> traceroute ipv6 2001:DB8::21:22
```

# track-port

Configures network reachability tracking for a specific Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRP-E) port.

## Syntax

**track-port** { **ethernet** *stackid**/**slot**/**port* | **lag** *lag-id* | **ve** *num* } [ **priority** *num* ]

**no track-port** { **ethernet** *stackid**/**slot**/**port* | **lag** *lag-id* | **ve** *num* } [ **priority** *num* ]

## Command Default

The network reachability of VRRP and VRRP-E ports or IPsec tunnels is not tracked.

## Parameters

**ethernet** *stackid slot port*

Configures network reachability tracking for a specific Ethernet interface. A forward slash "/" must be entered between the stackid, slot, and port numbers.

**lag** *lag-id*

Configures network reachability tracking for a specific LAG virtual interface. The LAG is identified by a decimal number.

**ve** *number*

Configures network reachability tracking for a virtual Ethernet interface. Valid values range from 1 through 255.

**priority** *num*

Sets the track priority. Valid numbers are from 1 through 254. The tracking priority number is used when a tracked interface up or down event is detected. For VRRP, if the tracked interface becomes disabled, the current router priority is reduced to the track-port priority. (For VRRP only, interface tracking does not have any effect on an owner router; the owner priority can not be changed under configuration from 255.) For VRRP-E, if the tracked interface becomes disabled, the current router priority is reduced by the track-port priority. For VRRP, the default is 2, and for VRRP-E, the default is 5.

## Modes

VRID interface configuration mode

## Usage Guidelines

This command can be used to track interfaces for VRRP or VRRP-E.

For VRRP, the tracked interface can be any valid Ethernet, or virtual Ethernet interface other than the one on which this command is issued. The maximum number of interfaces you can track per virtual router is 8.

Enter the **no track-port** command with the specified options to remove the tracked port configuration.

# Examples

The following example configures network reachability tracking on interface 1/1/6 and sets the track priority to 60.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.3/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# track-port ethernet 1/2/4 priority 60
```

# track-port (VSRP)

Configures the VRID on one interface to track the link state of another interface on the device.

## Syntax

**track-port** { **ethernet** *unit/slot/port* | **lag** *lag-id* | **ve** *number* } [ **priority** *number* ]

**no track-port** { **ethernet** *unit/slot/port* | **lag** *lag-id* | **ve** *number* } [ **priority** *number* ]

## Command Default

The VRID does not track an interface.

## Parameters

**ethernet** *unit/slot/port*
Configures the Ethernet interface to track.

| **lag** *lag-id*
Configures LAG virtual interface to track.

**ve** *number*
Configures the virtual Ethernet interface to track.

**priority** *number*
Changes the VSRP priority of the interface. The range is from 1 through 254.

## Modes

VSRP VRID configuration mode

## Usage Guidelines

Configuring this command is useful for tracking the state of the exit interface for the path for which the VRID is providing redundancy.

If the interface configured for tracking goes down, the VSRP VRID priority is reduced by the amount of the track port priority you specify.

The **priority** option changes the priority of the specified interface, overriding the default track port priority. To change the default track port priority, use the **backup track-priority** command.

The **no** form of the command removes the link state tracking.

## Examples

The following example configures the VRID to track an Ethernet interface .

```
device(config)# vlan 200
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# track-port ethernet 1/2/4
```

The following example configures the VRID to track a VE interface.

```
device(config)# vlan 200
device(config-vlan-200)# vsrp vrid 1
device(config-vlan-200-vrid-1)# track-port ve 4 priority 4
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | The command was modified to include the lag-id option. |

# traffic-policy count

Configures a traffic policy and enables counting the number of bytes and the conformance level per packet.

## Syntax

**traffic-policy** *traffic-policy-def* **count**

**no traffic-policy** *traffic-policy-def* **count**

## Command Default

No traffic policy is applied.

## Parameters

*traffic-policy-def*

Specifies the name of the traffic policy definition, in no more than seven alphanumeric characters.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of this command deletes a traffic policy definition.

## Examples

This example configures a traffic policy named TPD and enables counting of bytes and conformance levels.

```
device#configure terminal
device(config)#traffic-policy TPD count
```

# traffic-policy rate-limit adaptive

Configures an ACL-based flexible-bandwidth traffic policy to define rate limits on packets so that you can allow for bursts above the limit.

## Syntax

**traffic-policy** *traffic-policy-def* **rate-limit adaptive cir** *cir-value* **cbs** *cbs-value* **pir** *pir-value* **pbs** *pbs-value* **count**

**traffic-policy** *traffic-policy-def* **rate-limit adaptive cir** *cir-value* **cbs** *cbs-value* **pir** *pir-value* **pbs** *pbs-value* **exceed-action drop** [ **count** ]

**traffic-policy** *traffic-policy-def* **rate-limit adaptive cir** *cir-value* **cbs** *cbs-value* **pir** *pir-value* **pbs** *pbs-value* **exceed-action permit-at-low-pri** [ **count** | **remark-cos** [ **count** ] ]

**no traffic-policy** *traffic-policy-def* **rate-limit adaptive cir** *cir-value* **cbs** *cbs-value* **pir** *pir-value* **pbs** *pbs-value* **count**

**no traffic-policy** *traffic-policy-def* **rate-limit adaptive cir** *cir-value* **cbs** *cbs-value* **pir** *pir-value* **pbs** *pbs-value* **exceed-action drop** [ **count** ]

**no traffic-policy** *traffic-policy-def* **rate-limit adaptive cir** *cir-value* **cbs** *cbs-value* **pir** *pir-value* **pbs** *pbs-value* **exceed-action permit-at-low-pri** [ **count** | **remark-cos** [ **count** ] ]

## Command Default

No traffic policy is applied.

## Parameters

*traffic-policy-def*

Specifies the name of the traffic policy definition, in no more than seven alphanumeric characters.

**count**

Enables counting the number of bytes and the conformance level per packet. The single-rate three-color marker (srTCM) mechanism described in RFC 2697 is used.

**cir** *cir-value*

Specifies the committed information rate (CIR) in Kbps, that is, the guaranteed rate of inbound traffic that is allowed on a port. The range is 64 through 1,000,000 Kbps.

**cbs** *cbs-value*

Specifies the committed burst size (CBS), that is, the number of bytes per second allowed on a port before some packets exceed the CIR. You must specify a value greater than 0.

**pir** *pir-value*

Specifies the peak information rate (PIR) in Kbps, that is, the most inbound traffic that is allowed on a port. The *pir-value* must be equal to or greater than the *cir-value*.

**pbs** *pbs-value*

Specifies the peak burst size (PBS), that is, the most bytes per second allowed in a burst before all packets exceed the PIR. You must specify a value greater than 0.

**exceed-action**

Specifies the action for traffic that is more than is configured in the *cir-value* variable. If you do not configure this keyword, traffic that exceeds the *cir-value* is dropped

**drop**

Specifies dropping traffic that exceeds the rate limit.

**count**

Enables counting the number of bytes and the conformance level per packet. The two-rate three-color marker (trTCM) mechanism described in RFC 2698 is used.

**permit-at-low-pri**

Specifies permitting packets that exceed the *cir-value* and forward them at the lowest priority.

**remark-cos**

Sets the 802.1p priority of dropped packets to 0, that is, it sets the COS/PCP field value to 0 for the low priority traffic for any packet exceeding the rate limit set by the traffic policy

# Modes

Global configuration mode

# Usage Guidelines

The **no** form of this command deletes a traffic policy definition.

Traffic policies must be referenced by one or more ACLs before they can be effective. The policies are effective on ports to which the ACLs that reference them are bound.

**NOTE**

You cannot delete a traffic policy definition that a port is currently using. To delete a traffic policy, you must first unbind the associated ACL.

It is recommended that you specify a PBS value that is equal to or greater than the size of the largest possible IP packet in the stream.

# Examples

This example configures a traffic policy named TPDA4 that specifies a CIR of 10000 Kbps, a CBS of 1600 Kbps, a PIR of 20000 Kbps, and a PBS of 1000 Kbps a and dropping any traffic that exceeds those limits.

```
device# configure terminal
device(config)# traffic-policy TPDA4 rate-limit adaptive cir 10000 cbs 1600 pir 20000 pbs 4000 exceed-
action drop
```

# traffic-policy rate-limit fixed

Configures an ACL-based fixed-rate traffic policy to define rate limits on packets. It either drops all traffic that exceeds the limit, or forwards it at the lowest priority level.

## Syntax

**traffic-policy** *traffic-policy-def* **rate-limit fixed** *cir-value* **count**

**traffic-policy** *traffic-policy-def* **rate-limit fixed** *cir-value* **exceed-action drop** [ **count** ]

**traffic-policy** *traffic-policy-def* **rate-limit fixed** *cir-value* **exceed-action permit-at-low-pri** [ **count** | **remark-cos** [ **count** ] ]

**no traffic-policy** *traffic-policy-def* **rate-limit fixed** *cir-value* **count**

**no traffic-policy** *traffic-policy-def* **rate-limit fixed** *cir-value* **exceed-action drop** [ **count** ]

**no traffic-policy** *traffic-policy-def* **rate-limit fixed** *cir-value* **exceed-action permit-at-low-pri** [ **count** | **remark-cos** [ **count** ] ]

## Command Default

No traffic policy is applied.

## Parameters

*traffic-policy-def*
Specifies the name of the traffic policy definition, in no more than seven alphanumeric characters.

*cir-value*
Specifies the committed information rate (CIR) in Kbps, that is, the guaranteed rate of inbound traffic that is allowed on a port. The range is 64 through 1,000,000 Kbps.

**count**
Enables counting the number of bytes and the conformance level per packet. The single-rate three-color marker (srTCM) mechanism described in RFC 2697 is used.

**exceed-action**
Specifies the action for traffic that is more than is configured in the *cir-value* variable. If you do not configure this keyword, traffic that exceeds the *cir-value* is dropped

**drop**
Specifies dropping traffic that exceeds the rate limit.

**count**
Enables counting the number of bytes and the conformance level per packet. The single-rate three-color marker (srTCM) mechanism described in RFC 2697 is used.

**permit-at-low-pri**
Specifies permitting packets that exceed the *cir-value* and forward them at the lowest priority.

**remark-cos**
Sets the 802.1p priority of dropped packets to 0, that is, it sets the COS/PCP field value to 0 for the low priority traffic for any packet exceeding the rate limit set by the traffic policy

# Modes

Global configuration mode

# Usage Guidelines

The **no** form of this command deletes a traffic policy definition.

Traffic policies must be referenced by one or more ACLs before they can be effective. The policies are effective on ports to which the ACLs that reference them are bound.

**NOTE**
You cannot delete a traffic policy definition that is currently in use on a port. To delete a traffic policy, you must first unbind the associated ACL.

# Examples

This example configures a traffic policy named TPD1 that specifies a CIR of 100 Kbps and dropping any traffic that exceeds the limit.

```
device# configure terminal
device(config)# traffic-policy TPD1 rate-limit fixed 100 exceed-action drop
```

# transform

Configures a transform set for an IPsec proposal.

## Syntax

**transform esp**

## Command Default

Encapsulating Security Payload (ESP)

## Parameters

**esp**

Specifies the Encapsulating Security Payload transform set.

## Modes

IPsec proposal configuration mode

## Usage Guidelines

Only ESP is currently supported. Therefore, you do not need to configure the transform set for an IPsec proposal because the only option is configured by default.

## Examples

The following example shows how to configure ESP as the transform set for an IPsec proposal named ipsec_prop.

```
device(config)# ipsec proposal ipsec_prop
device(config-ipsec-proposal-ipsec_prop)# transform esp
```

## History

| Release version | Command history |
|---|---|
| 8.0.50 | This command was introduced. |

# trunk-threshold

Configures the threshold value for the number of active member ports in a LAG, below which all the ports in a LAG group are disabled.

## Syntax

**trunk-threshold** *number*

**no trunk-threshold** *number*

## Command Default

The t runk threshold is set to 1.

## Parameters

*number*

Specifies the number of ports as the threshold number. You can specify a threshold from 1 (the default) up to the number of ports in the LAG group.

## Modes

LAG configuration mode

## Usage Guidelines

When a LAG is shut down because the number of ports drops below the configured threshold, the LAG is kept intact and it is re-enabled if enough ports become active to reach the threshold.

**NOTE**
The **trunk-threshold** command cannot be used in conjunction with protected link groups.

**NOTE**
The **trunk-threshold** command is only applicable for the configuration of static LAGs.

The **trunk-threshold** command should be configured only at one end of the LAG. If it is set on both ends, link failures will result in race conditions and the LAG not function properly. Use a short LACP timeout when setting the **trunk-threshold** value equal to the number of links in the LAG or connecting to third-party devices.

The **no** form of the command removes the **trunk-threshold** configuration.

# Examples

The following example shows how to establish a LAG group consisting of four ports, and then establish a threshold for this LAG group of three ports. If the number of active ports drops below three, then all the ports in the LAG group are disabled.

```
device(config)# lag blue static
device(config-lag-blue)# ports ethernet 1/3/1 to 1/3/4
device(config-lag-blue)# trunk-threshold 3
```

# trust dscp

Configures the device to honor DSCP-based QoS for routed and switched traffic.

## Syntax

**trust dscp**

**no trust dscp**

## Command Default

The interface honors the Layer 2 CoS value.

## Modes

Interface configuration mode

## Usage Guidelines

The **no** form of the command disables the device from honoring DSCP-based QoS.

**NOTE**
This command is not supported with 802.1p priority override.

## Examples

The following example honors DSCP-based QoS.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# trust dscp
```

# trust-port

Configures ports of a Web Authentication VLAN as trusted ports.

## Syntax

**trust-port ethernet** *stack/slot/port* [**to** *stack/slot/port* ]

**no trust-port ethernet** *stack/slot/port* [**to** *stack/slot/port* ]

## Command Default

Ports of a Web Authentication VLAN are not trusted.

## Parameters

**ethernet** *stack/slot/port*
Configures the specified Ethernet interface as a trusted port.

**to** *stack/slot/port*
Configures a range of Ethernet interfaces as trusted.

## Modes

Web Authentication configuration mode

## Usage Guidelines

All hosts connected to the trusted ports need not authenticate and are automatically allowed access to the network.

The **no** form of the command removes the trusted port configuration.

## Examples

The following example shows how to configure an Ethernet interface as a trusted port.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# trust-port ethernet 1/1/1
```

The following example shows how to configure a range of ports as trusted.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# trust-port ethernet 1/1/1 to 1/1/10
```

# tunnel destination

Configures the destination address for a specific tunnel interface.

## Syntax

**tunnel destination** { *ip-address* | *ipv6-address* }

**no tunnel destination** { *ip-address* | *ipv6-address* }

## Command Default

No tunnel interface destination is configured.

## Parameters

*ip-address*
Specifies the IPv4 address of an interface.

*ipv6-address*
Specifies the IPv6 address of the destination.

## Modes

Interface tunnel configuration mode

## Usage Guidelines

ICX 7150 devices do not support tunnels.

You must ensure that a route to the tunnel destination exists on the tunnel source device and create a static route if necessary.

The **no** form of the command removes the configured destination for the tunnel interface.

## Examples

The following example configures the IP address 10.1.2.3 as the destination address for a specific tunnel interface.

```
device# configure terminal
device(config)# interface tunnel 3
device(config-tnif-3)# tunnel destination 10.1.2.3
```

## History

| Release version | Command history |
|---|---|
| 08.0.41 | This command was introduced. |
| 08.0.70 | This command added support for IPv6 addressing for IPsec tunnels. |

## Related Commands

tunnel source

# tunnel mode gre ip

Enables generic routing encapsulation (GRE) over on a tunnel interface and specifies that the tunneling protocol is IPv4.

## Syntax

**tunnel mode gre ip**

**no tunnel mode gre ip**

## Command Default

GRE is disabled.

## Modes

Interface tunnel configuration mode

## Usage Guidelines

ICX 7150 devices do not support tunnels.

Use the **no tunnel mode gre ip** command to disable the GRE IP tunnel encapsulation method for the tunnel interface.

## Examples

The following example enables GRE IP encapsulation on a tunnel interface.

```
device# configure terminal
device(config)# interface tunnel 3
device(config-tnif-3)# tunnel mode gre ip
```

## Related Commands

interface tunnel

# tunnel mode ipsec

Configures the mode of a virtual tunnel interface (VTI) as IPsec.

## Syntax

**tunnel mode ipsec** { **ipv4** | **ipv6** }

**no tunnel mode ipsec** { **ipv4** | **ipv6** }

## Command Default

The tunnel mode is not configured for a VTI.

## Parameters

**ipv4**

Specifies the application of IPsec protection for IPv4 packets transmitted over the tunnel.

**ipv6**

Specifies the application of IPsec protection for IPv6 packets transmitted over the tunnel.

## Modes

Tunnel interface configuration mode

## Usage Guidelines

IPsec is supported only on ICX 7450 devices.

ICX 7150 devices do not support tunnels.

The **no** form of the command removes the IPsec mode configuration for the VTI.

## Examples

The following example shows how to set the mode for tunnel 1 to IPsec for IPv4 traffic.

```
device# configure terminal
device(config) interface tunnel 1
device(config-tnif-1)# tunnel mode ipsec ipv4
```

## History

| Release version | Command history |
|---|---|
| 08.0.41 | This command was introduced. |
| 08.0.70 | This command added support for IPv6 IPsec tunnels. |

# tunnel mode ipv6ip

Configures the tunnel mode as a manual IPv6 tunnel.

## Syntax

**tunnel mode ipv6ip**

**no tunnel mode ipv6ip**

## Command Default

A tunnel is not configured.

## Modes

Interface tunnel configuration mode

## Usage Guidelines

You can use a manually configured tunnel to connect two isolated IPv6 domains. You should deploy this point-to-point tunneling mechanism if you need a permanent and stable connection.

ICX 7150 devices do not support tunnels.

The **no** form of the command removes the configured tunnel mode.

## Examples

The following example configures the tunnel mode as a manual IPv6 tunnel.

```
device(config)# interface tunnel 1
device(config-tnif-1)# tunnel source ethernet 1/1/1
device(config-tnif-1)# tunnel destination 10.162.100.1
device(config-tnif-1)# tunnel mode ipv6ip
device(config-tnif-1)# ipv6 enable
```

# tunnel path-mtu-discovery

Enables Path MTU Discovery (PMTUD).

## Syntax

**tunnel path-mtu-discovery** { **age-timer** { *time* | **infinite** } | **disable** }

**no tunnel path-mtu-discovery** { **age-timer** { *time* | **infinite** } | **disable** }

## Command Default

PMTUD is enabled by default.

## Parameters

**age-timer**

Configures the time after which the path MTU resets to its original value.

*time*

Sets the time after which the path MTU resets to its original value. Valid values are 10 to 30 minutes. The default value is 10 minutes.

**infinte**

Sets the aging time as infinite, that is, disables aging for PMTUD.

**disable**

Disables aging for PMTUD.

## Modes

Tunnel interface configuration mode

## Usage Guidelines

ICX 7150 devices do not support tunnels.

The **no** form of the command disables PMTUD and resets the aging to the default value of 10 minutes.

## Examples

The following example changes the reset time (default age timer) to a value of 25.

```
device(config)# tunnel interface 1
device(config-tnif-1)# tunnel path-mtu-discovery age-timer 25
```

The following example disables aging for PMTUD.

```
device(config)# tunnel interface 1
device(config-tnif-1)# tunnel path-mtu-discovery disable
```

# tunnel protection ipsec profile

Configures an IPsec profile for an IPsec virtual tunnel interface (VTI).

## Syntax

**tunnel protection ipsec profile** *ipsec-profile-name*

**no tunnel protection ipsec profile** *ipsec-profile-name*

## Command Default

An IPsec profile is not configured for the VTI.

## Parameters

*ipsec-profile-name*

Specifies the name of the IPsec profile to secure packets that go out on this interface.

## Modes

Interface configuration mode

## Usage Guidelines

Before executing this command, the tunnel mode must be set to ipsec by using the **tunnel mode ipsec** command.

IPsec is supported only on ICX 7450 devices.

ICX 7150 devices do not support tunnels.

The **no** form of the command removes the IPsec profile configuration.

## Examples

The following example shows how to configure an IPsec profile named prof_blue on a VTI with the tunnel ID 1 for an IPsec IPv4 tunnel.

```
device# configure terminal
device (config)# interface tunnel 1
device(config-tnif-1)# tunnel mode ipsec ipv4
device(config-tnif-1)# tunnel protection ipsec profile prof_blue
```

## History

| Release version | Command history |
|---|---|
| 08.0.41 | This command was introduced. |

# tunnel source

Configures the source address or a source interface for a specific tunnel interface.

## Syntax

**tunnel destination** { *ip address* | *ipv6 address* | **ethernet** *unit / slot / port* | **loopback** *number* | **ve** *vlan_id* }

**no tunnel destination** { *ip address* | *ipv6 address* | **ethernet** *unit / slot / port* | **loopback** *number* | **ve** *vlan_id* }

## Command Default

No source address or interface is configured.

## Parameters

*ip address*
Specifies the IPv4 address of an interface.

*ipv6 address*
Specifies the IPv6 address of the source.

**ethernet** *unit / slot / port*
Specifies an Ethernet interface.

**loopback** *number*
Specifies an loopback port.

**ve** *vlan_id*
Specifies a VE interface.

## Modes

Interface tunnel configuration mode

## Usage Guidelines

Use the **no tunnel source** command to remove the configured source for the tunnel interface.

The tunnel source address should be one of the router IP addresses configured on a physical, loopback, or VE interface, through which the other end of the tunnel is reachable. The source interface must have at least one IP address configured on it.

ICX 7150 devices do not support tunnels.

## Examples

The following example configures the IP address 10.1.2.4 as the source address for a specific tunnel interface.

```
device# configure terminal
device(config)# interface tunnel 3
device(config-tnif-3)# tunnel source 10.1.2.4
```

The following example sets an Ethernet interface as a source tunnel.

```
device# configure terminal
device(config)# interface tunnel 1
device(config-tnif-1)# tunnel source ethernet 1/3/1
```

# History

| Release version | Command history |
|---|---|
| 08.0.41 | This command was introduced. |
| 08.0.70 | This command added support for IPv6 addressing for IPsec tunnels. |

# Related Commands

tunnel destination

# tunnel tos

Configures the Type of Service (ToS) value for an IPsec virtual tunnel interface (VTI).

## Syntax

**tunnel tos** *tos*

**no tunnel tos** *tos*

## Command Default

The Type of Service is not configured for the IPsec VTI.

## Parameters

*tos*

Specifies the Type of Service (ToS) value. The range is from 0 through 255.

## Modes

Tunnel interface configuration mode

## Usage Guidelines

When ToS is not configured for an IPsec VTI, the ToS value that is configured on the inner IP header is copied to the outer IP header.

ToS configuration is only supported on IPsec tunnel interfaces. The mode of the VTI must be set to **ipsec** before executing this command.

ICX 7150 devices do not support tunnels.

The **no** form of the command removes the ToS configuration on the VTI.

## Examples

The following example shows how to configure a ToS value of 3 for an IPsec tunnel identified as 1.

```
device(config)# interface tunnel 1
device(config-tnif-1)# tunnel mode ipsec ipv4
device(config-tnif-1)# tunnel tos 3
```

## History

| Release version | Command history |
|---|---|
| 8.0.41 | This command was introduced. |

# tunnel vrf

Configures the base VRF for an IPsec virtual tunnel interface (VTI).

## Syntax

**tunnel vrf** *name*

**no tunnel vrf** *name*

## Command Default

The default VRF is the base VRF for the IPsec VTI.

## Parameters

*name*

Specifies the name of the base VRF.

## Modes

Tunnel interface configuration mode

## Usage Guidelines

Configuration of a base VRF is only supported on IPsec tunnel interfaces. The mode of the VTI must be set to **ipsec** before executing this command.

ICX 7150 devices do not support tunnels.

The **no** form of the command removes the base VRF configuration for the VTI.

## Examples

The following example shows how to configure a VRF named blue as the base VRF for an IPsec tunnel identified as 1.

```
device(config)# interface tunnel 1
device(config-tnif-1)# tunnel mode ipsec ipv4
device(config-tnif-1)# tunnel vrf blue
```

## History

| Release version | Command history |
|---|---|
| 8.0.41 | This command was introduced. |

# unknown-unicast limit (enable)

Configures the maximum number of unknown unicast packets allowed per second.

## Syntax

**unknown-unicast limit** *num* **kbps**

**no unknown-unicast limit** *num* **kbps**

## Command Default

Unknown unicast rate limiting is disabled.

## Parameters

*num*

Specifies the maximum number of unknown unicast packets per second. The value can be 1 to 8388607.

**kbps**

Enables byte-based limiting. The value can be 1 to Max Port Speed.

## Modes

Interface configuration mode

## Usage Guidelines

Use 0 or the **no** form of the command to disable limiting.

## Examples

The following example enables a unknown unicast limit of 131072 kbps.

```
device(config)# interface ethernet 9/1/1
device(config-if-e1000-9/1/1)# unknown-unicast limit 131072 kbps
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | The command was introduced. |

# unknown-unicast limit (logging)

Enables Syslog logging of unknown unicast packets.

## Syntax

**unknown-unicast limit** *num* **kbps** [ **log** ]

**no unknown-unicast limit** *num* **kbps** [ **log** ]

## Command Default

Unknown unicast rate logging is disabled.

## Parameters

*num*

Specifies the maximum number of packets per second. The value can be any 1 to 8388607.

**kbps**

Enables byte-based limiting. The value can be 1 to Max Port Speed.

**log**

Enables Syslog logging when the unknown unicast limit exceeds *num* **kpbs** .

## Modes

Interface configuration mode

## Usage Guidelines

Use 0 or the **no** form of the command to disable limiting.

## Examples

The following example enables unknown unicast logging when the configured limit exceeds 100 Kbps.

```
device(config)# interface ethernet 1/2/1
device(config-if-e1000-1/2/1)# unknown-unicast limit 100 kbps log
```

## History

| Release version | Command history |
|---|---|
| 8.0.10 | The command was introduced. |
| 8.0.40a | The command was modified to include the keyword **log**. |

# unmount disk0

Unmounts the external USB.

## Syntax

**unmount disk0**

## Modes

User EXEC mode.

## Examples

The following example unmounts the external USB.

```
device# unmount disk0
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# untagged

Adds untagged ports to the VLAN.

## Syntax

**untagged** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

**no untagged** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] }

## Parameters

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Configures and adds a port, set of ports, or range of ports as untagged.

**lag** *lag-id* [ **to** *lag-id* ]

Configures a LAG virtual interface, set of LAG virtual interfaces, or range of LAG virtual interfaces to be added as untagged ports. (LAG ID is a decimal value.)

**to**

When followed by a port number, onfigures a range of ports. When followed by a LAG ID, configures a range of LAGs.

## Modes

VLAN configuration mode

## Usage Guidelines

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of the command removes the untagged ports on the VLAN.

## Examples

The following example shows how to add a range of untagged Ethernet ports to a port-based VLAN.

```
device(config)# vlan 222 by port
device(config-vlan-222)# untagged ethernet 1/1/1 to 1/1/8
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was modified to include LAG ID options. |

# update-lag-name

Changes the name of an existing LAG without causing any impact on the functionality of the LAG.

## Syntax

**update-lag-name** *new-name*

## Parameters

*new-name*

Specifies the new name for the LAG.

## Modes

LAG configuration mode

## Usage Guidelines

The new name must be unique and unused.

## Examples

The following example renames LAG blue to blue1.

```
device(config)# lag blue static
device(config-lag-blue)# update-lag-name blue1
INFORMATION: Lag blue with ID 1 is updated to new name blue1
device(config)#
```

## History

| Release version | Command history |
|---|---|
| 08.0.30 | This command was introduced. |

# update-time (BGP)

Configures the interval at which BGP next-hop tables are modified. BGP next-hop tables should always have IGP (non-BGP) routes.

## Syntax

**update-time** *sec*

**no update-time** *sec*

## Parameters

*sec*

Update time in seconds. Valid values range from 0 through 30. Default is 5 seconds.

## Modes

BGP configuration mode

BGP address-family IPv6 unicast configuration mode

## Usage Guidelines

When this command is entered in BGP global configuration mode, it applies only to the IPv4 address family. Use this command in BGP address-family IPv6 unicast configuration mode for BGP4+ configurations.

The update time determines how often the device computes the routes (next-hops). Lowering the value set by the **update-time** command increases the convergence rate.

By default, the device updates the BGP next-hop tables and affected BGP routes five seconds following IGP route changes. Setting the update time value to 0 permits fast BGP convergence for situations such as a link failure or IGP route changes, starting the BGP route calculation in sub-second time.

**NOTE**
Use the **advertisement-interval** command to determine how often to advertise IGP routes to the BGP neighbor.

## Examples

The following example permits fast convergence for the IPv4 unicast address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# update-time 0
```

The following example sets the update time interval to 30 the IPv6 unicast address family.

```
device# configure terminal
device(config)# router bgp
device(config-bgp-router)# address-family ipv6 unicast
device(config-bgp-ipv6u)# update-time 30
```

# update-time (RIP)

Specifies how often the device sends RIP route advertisements to its RIP neighbors.

## Syntax

**update-time** *value*

**no update-time** *value*

## Command Default

By default, the update interval is 30 seconds.

## Parameters

*value*

Specifies the update interval in seconds. Allowable values are from 3 through 21845.

## Modes

RIP router configuration mode

## Usage Guidelines

The **no** form of the command returns the update interval to its default value.

The update time can also be modified through the RIP **timers** command.

## Examples

The following example configures the RIP router to send route advertisements to its neighbors every two minutes (120 seconds).

```
device# configure terminal
device(config)# router rip
device(config-rip-router)# update-time 120
```

# use-radius-server

Maps a RADIUS server to a port.

## Syntax

**use-radius-server** *ip-address*

**no use-radius-server** *ip-address*

## Command Default

The RADIUS server is not mapped to any port.

## Parameters

*ip-address*
The IP address of the RADIUS server.

## Modes

Interface configuration mode

## Usage Guidelines

Once the RADIUS server is mapped to a port, the port sends the RADIUS request to the configured RADIUS server.

The **no** form of the command removes the mapping of the RADIUS server to the port.

## Examples

The following example shows how to map a RADIUS server to the interface 1/1/3 (port 3). Port 3 sends a RADIUS request to 10.10.10.103 first, because it is the first server mapped to the port. If the request fails, the server will go to 10.10.10.110.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# use-radius-server 10.10.10.103
device(config-if-e1000-1/1/1)# use-radius-server 10.10.10.110
```

# use-v2-checksum

Enables the v2 checksum computation method for an IPv4 Virtual Router Redundancy Protocol version 3 (VRRPv3) session.

## Syntax

**use-v2-checksum**

**no use-v2-checksum**

## Command Default

VRRPv3 uses the v3 checksum computation method.

## Modes

VRRP configuration mode

## Usage Guidelines

The **no** form of this command enables the default v3 checksum computation method in VRRPv3 sessions.

Some non-Ruckus devices only use the v2 checksum computation method in VRRPv3. This command enables the v2 checksum computation method in VRRPv3 and provides interoperability with these non-Ruckus devices.

## Examples

The following example shows the v2 checksum computation method enabled in IPv4 and IPv6 VRRPv3 instances.

```
device(config)# interface ve 3
device(config-vif-3)# ipv4 vrrp vrid 2
device(config-vif-3-vrid-2)# version v3
device(config-vif-3-vrid-2)# use-v2-checksum

device(config)# interface ve 3
device(config-vif-3)# ipv6 vrrp vrid 2
device(config-vif-3-vrid-2)# use-v2-checksum
```

## History

| Release version | Command history |
|---|---|
| 08.0.01 | This command was introduced for IPv6 VRRPv3 sessions running on FastIron device images. |
| 08.0.10b | This command was modified for IPv4 VRRPv3 sessions running on FastIron device images. |

# use-vrrp-path (RIP)

Suppresses RIP advertisements for interfaces on which Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRP-E) backup routers are configured.

## Syntax

**use-vrrp-path**

**no use-vrrp-path**

## Command Default

RIP advertisements are enabled.

## Modes

RIP configuration mode

## Usage Guidelines

The command applies only to devices configured for Virtual Router Redundancy Protocol (VRRP) or for VRRP Extended (VRRPE). The same command syntax is used for both protocols. The command applies only if you have specified an IP address to back up and is valid only on Layer 3 Switches.

Normally for Layer 3, a VSRP backup includes route information in RIP advertisements for an interface with a VRRP or VRRP-E backup. As a result, other Layer 3 switches receive multiple paths for the backed-up interface and may sometimes unsuccessfully use the path to the backup router rather than the path to the master.

Use the command to suppress RIP advertisements from the backup router on the interface. This ensures that the interface advertises paths to the master router only.

The **no** form of this command resets the default behavior, and the interface sends RIP advertisements from the backup router.

## Examples

The following example shows how to suppress RIP advertisements from backup VRRP or VRRP-E routers.

```
device(config)# router rip
device(config-rip-router)# use-vrrp-path
```

# username

Creates or updates a user account.

## Syntax

**username** *username-string* { [ **privilege** *privilege-level* ] { **password** *password-string* | **create-password** *password-string* | **nopassword** } | **access-time** *begin-time* **to** *end-time* | **enable** | **expires** *days* }

**no username** *username-string* [ [ **privilege** *privilege-level* ] { **password** *password-string* | **create-password** *password-string* | **nopassword** } | **access-time** *begin-time* **to** *end-time* | **enable** | **expires** *days* ]

## Command Default

The user account is not created.

## Parameters

*username-string*

The configured username. You can enter up to 48 characters.

**privilege** *privilege-level*

Sets the user's privilege level. The default privilege level is 0. You can specify one of the following levels:

**0**

Super User level (full read-write access).

**4**

Port Configuration level.

**5**

Read Only level.

**password** *password-string*

Configures the password for the user. You can enter up to 48 characters.

**create-password** *password-string*

Creates an encrypted password for the user. You can enter up to 48 characters.

**nopassword**

Configures the user login without a password.

**access-time** *begin-time* **to** *end-time*

Configures the access permission for a specified period of time of the day, that is, between the specified beginning access time and ending access time.

**enable**

Enables the user for login access after the login access is disabled.

**expires** *days*

Configures the password expiration time in days. The valid values are from 1 through 365.

# Modes

Global configuration mode

# Usage Guidelines

You must be logged in with Super User access (privilege level 0) to add or delete user accounts or configure or modify other access parameters.

By default, user account details can be deleted or modified without any authentication. Unauthorized deletion or modification of the user account can be prevented using the **service local-user-protection** command. If the user account security is enabled using the **service local-user-protection** command, deletion of user accounts or changing the password or privilege level of the user is permitted only upon successful validation of the existing user password.

If the **enable strict password enforcement** command is enabled on the device, for the password string, you must enter a minimum of eight characters containing the following combinations:

- At least two uppercase characters
- At least two lowercase characters
- At least two numeric characters
- At least two special characters

You can use the **show user** command to display the user account details.

The **no** form of the command removes the user or the other user parameters.

# Examples

The following example configures the privilege level of Super User access (0) for a user.

```
device(config)# username user1 privilege 0 password *******
```

The following example configures an unencrypted password for a user.

```
device(config)# username user1 password xpassx
```

The following example configures an encrypted password for a user.

```
device(config)# username user1 create-password xpassx
```

The following example creates a user account without a password.

```
device(config)# username user1 nopassword
```

The following example configures the access time for a user.

```
device(config)# username user1 access-time 00:00:00 to 12:00:00
```

The following example enables a user account if it is disabled.

```
device(config)# username user1 enable
```

The following example sets the user password to expire in 30 days.

```
device(config)# username user expires 30
```

The following example prompts the user to confirm existing password before successful password modification.

```
device(config)# username user1 password xpassx
device(config)# service local-user-protection
device(config)# username user1 password ypasswordy
User already exists. Do you want to modify: (enter 'y' or 'n'): y
To modify or remove user, enter current password: ******
```

# History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was modified to prompt the user to enter a valid password before deleting a user account or modifying the password or privilege level of the user. |

# username (Local database)

Creates a user record in the local user database.

## Syntax

**username** *username* **password** *password-string*

**no username** *username* [ **password** *password-string* ]

## Command Default

User records are not created.

## Parameters

*username*

Specifies the username for the user as an ASCII string. You can specify up to 31 characters.

**password** *password-string*

Specifies the password for the user. You can specify up to 29 characters.

## Modes

Local user database configuration mode

## Usage Guidelines

You can add up to 30 usernames and passwords to a local user database.

The **no** form of the command removes the user record from the local user database.

## Examples

The following example creates a new user account and adds it to a local user database.

```
device(config)# local-userdb userdb1
device(config-localuserdb-userdb1)# username XYZ password A5!fk3p
```

# vendor-class

Specifies the vendor type (option 60) and configuration value for a DHCP client.

## Syntax

**vendor-class** { **ascii** } *ascii string*

## Parameters

**ascii**

Specifies the ascii keyword.

*ascii string*

Specifies the ASCII string value of the DHCP client.

## Modes

DHCP server pool configuration mode

## Examples

The following example specifies option 60 using the ASCII option for a Ruckus AP.

```
device# configure terminal
device(config)# ip dhcp-server-pool ruckus
device(ip dhcp-server pool ruckus)#    vendor-class ascii "Ruckus CPE"
device(ip dhcp-server pool ruckus)# deploy
```

## History

| Release version | Command history |
|---|---|
| 08.0.30mb | An additional example was added in the Examples section for option 60. |
| 08.0.61 | Support for this command was added. |

# verify

Allows the verification of boot images based on hash codes and the generation of hash codes where needed.

## Syntax

**verify** { **md5** | **sha1** | **crc32** } { **primary** | **secondary** } [ *string* ]

## Parameters

**md5**
Verifies the file content using an MD5 checksum and generates a 16-byte hash code.

**sha1**
Verifies the file content using SHA-1 and generates a 20-byte hash code.

**crc32**
Verifies the file content using CRC32 and generates a 4-byte hash code.

**primary**
Verifies the primary boot image.

**secondary**
Verifies the secondary boot image.

*string*
A valid image file name or a generated hash code value.

## Modes

Privileged EXEC mode

## Usage Guidelines

This feature lets you select from three data integrity verification algorithms:

- MD5: Message-digest algorithm (RFC 1321)
- SHA1: US Secure Hash Algorithm (RFC 3174)
- CRC: Cyclic redundancy check algorithm

## Examples

The following example shows how the **verify** command can be used to generate an MD5 hash value for the secondary image.

```
device# verify md5 secondary
device#.......................Done
Size = 2044830, MD5 01c410d6d153189a4a5d36c955653862
```

The following example shows how the **verify** command can be used to generate a SHA-1 hash value for the secondary image.

```
device# verify sha1 secondary
device#.......................Done
Size = 2044830, SHA1 49d12d26552072337f7f5fcaef4cf4b742a9f525
```

The following example shows how the **verify** command can be used to generate a CRC32 hash value for the secondary image.

```
device# verify crc32 secondary
device#.......................Done
Size = 2044830, CRC32 b31fcbc0
```

The following example shows how the **verify** command can be used to verify the hash value of a secondary image with a known value.

```
device# verify md5 secondary 01c410d6d153189a4a5d36c955653861
device#.......................Done
Size = 2044830, MD5 01c410d6d153189a4a5d36c955653861
Verification SUCEEDED.
```

The following example shows how the **verify** command can be used to verify the SHA-1 hash value of a secondary image with a known value.

```
device# verify sha1 secondary 49d12d26552072337f7f5fcaef4cf4b742a9f525
device#.......................Done
Size = 2044830, sha 49d12d26552072337f7f5fcaef4cf4b742a9f525
Verification SUCCEEDED.
```

The following example shows how the **verify** command can be used to verify the CRC32 hash value of a secondary image with a known value.

```
device# verify crc32 secondary b31fcbc0
device#.......................Done
Size = 2044830, CRC32 b31fcbc0
Verification SUCCEEDED
```

# version

Sets the version number for a Virtual Router Redundancy Protocol (VRRP) session.

## Syntax

**version** { **v2** | **v3** }

**no version** { **v2** | **v3** }

## Command Default

VRRP version 2 is the default.

## Parameters

**v2**

Configures VRRP version 2 for this session.

**v3**

Configures VRRP version 3 for this session.

## Modes

Virtual routing ID interface configuration mode

## Usage Guidelines

The **no** form of this command resets the VRRP session to the default of version 2.

VRRP version 2 supports IPv4 addresses, and VRRP version 3 supports both IPv4 and IPv6 addresses.

**NOTE**
Mixed mode (VRRPv2 and VRRPv3) is not supported in the same VRRP virtual routing ID (VRID) session.

## Examples

The following example sets VRRP routing instance VRID 1 to version 3.

```
device# configure terminal
device(config)# router vrrp
device(config)# interface ethernet 1/1/6
device(config-if-e1000-1/1/6)# ip address 10.53.5.1/24
device(config-if-e1000-1/1/6)# ip vrrp vrid 1
device(config-if-e1000-1/1/6-vrid-1)# version v3
```

# violation

Configures the action that must be taken according to the configurable violation modes when a security violation occurs.

## Syntax

**violation** { **protect** | **restrict** *age* | **shutdown** *time* }

## Command Default

The default action upon PMS violation is protect.

## Parameters

**protect**

Configures the device to drop all packets which are not from secure MAC addresses. In the protect mode, the port never gets shut down.

**restrict**

Configures the device to drop packets from violated address and allow packets from secure addresses.

*age*

Configures the time, in minutes, for which the device drops packets after which the violated MAC address is aged out. The valid values are from 0 through 1440 minutes. The default is 5 minutes. Specifying 0 drops packets from the violating address permanently.

**shutdown** *time*

Configures the device to disable the port upon detection of first violated MAC address. The valid values are from 0 through 1440 minutes. The default value is 0 which shuts down the port permanently when a security violation occurs. The shutdown time which serves as a recovery interval, brings up the port within a configured time without any manual intervention.

## Modes

Port security configuration mode

Port security interface configuration mode

## Usage Guidelines

A security violation occurs when a user tries to connect to a port where a MAC address is already locked, or the maximum limit for the number of secure MAC addresses allowed on the interface is exceeded. When a security violation occurs, an SNMP trap and syslog message are generated.

When the **restrict** option is used, maximum number of MAC addresses that can be restricted is 128. If the number of violated MAC addresses exceeds 128, the port will be shut down. In this mode, manual intervention is required to bring up the port that is forced to shut down after the security violation. Aging for restricted MAC addresses is done in software. There can be a worst case inaccuracy of one minute from the specified time. The restricted MAC addresses are denied in hardware.

The required action must be specified to switch between PMS violation modes.

# Examples

The following example configures the violation mode as protect that, upon security violation, drops all packets which are not from secure MAC addresses.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port security
device(config-port-security-e1000-1/1/1)# violation protect
```

The following example configures the device to drop packets from a violating address and allow packets from secure addresses.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port security
device(config-port-security-e1000-1/1/1)# violation restrict
```

The following example configures the number of minutes that the device drops packets from a violating address.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port security
device(config-port-security-e1000-1/1/1)# violation restrict 10
```

The following example shuts downs the port for 5 minutes when a security violation occurs.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# port security
device(config-port-security-e1000-1/1/1)# violation shutdown 5
```

# History

| Release version | Command history |
|---|---|
| 08.0.70 | This command was modified to add the **protect** option. |

# virtual-ip

Configures the IP address of the external captive portal server as the virtual IP address.

## Syntax

**virtual-ip** { *ip-address* | *ASCII string* }

**no virtual-ip** { *ip-address* | *ASCII string* }

## Command Default

A virtual IP address is not configured.

## Parameters

*ip-address*
Specifies the IP address of the external captive portal server where the web pages are hosted.

*ASCII string*
Specifies the server name of the external captive portal server where the web pages are hosted.

## Modes

Captive portal configuration mode

## Usage Guidelines

The **no** form of the command removes the virtual IP address configuration.

## Examples

The following example configures the IP address of the external captive portal server as the virtual IP address.

```
device(config)# captive-portal cp_ruckus
device(config-cp-cp_ruckus)# virtual-ip 10.21.240.42
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j |

# virtual-port

Configures the HTTP port number to facilitate HTTP services for the clients in external Web Authentication.

## Syntax

**virtual-port** *http-port-number*

**no virtual-port** *http-port-number*

## Command Default

A virtual port number is not configured.

## Parameters

*http-port-number*

Specifies the port number. By default, HTTPS is used and the default port number for HTTPS is 443.

## Modes

Captive portal configuration mode

## Usage Guidelines

The protocol configured in the Captive Portal profile must be the same as the protocol configured as part of web management access using the **web-management** command.

You can also specify HTTP mode and the default port number for HTTP is 80.

The **no** form of the command removes the virtual port number configuration.

## Examples

The following example configures the virtual port number used by HTTP.

```
device(config)# captive-portal cp_ruckus
device(config-cp-cp_ruckus)# virtual-port 80
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |
| 8.0.30j | This command was added to FastIron 8.0.30j |

# vlan

Creates VLANs.

## Syntax

**vlan** *vlan-id* [ **to** *vlan-id* | [ *vlan-id* **to** *vlan-id* | *vlan-id* ] ... ] [ **name** *string* ] [ **by port** ]

**no vlan** *vlan-id* [ **to** *vlan-id* | [ *vlan-id* **to** *vlan-id* | *vlan-id* ] ... ] [ **name** *string* ] [ **by port** ]

## Command Default

The default VLAN is 1. Maximum allowed discrete or set of VLAN(s) is 1024.

## Parameters

*vlan-id*
: Specifies the VLAN ID.

**to** *vlan-id*
: Creates a range of VLANs.

**name** *string*
: Specifies the name of the VLAN. The name can be up to 32 characters in length.

**by port**
: Configures the VLAN as a port-based VLAN.

## Modes

Global configuration mode

## Usage Guidelines

You can configure up to 1023 port-based VLANs on a device running Layer 2 code or 4061 port-based VLANs on a device running Layer 3 code. Each port-based VLAN can contain either tagged or untagged ports. A port cannot be a member of more than one port-based VLAN unless the port is tagged.

**NOTE**
VLAN IDs 4087, 4090, and 4093 are reserved for Ruckus internal use only. VLAN 4094 is reserved for use by Single STP. Also, VLAN IDs 4091 and 4092 may be reserved for Ruckus internal use only. If you want to use VLANs 4091 and 4092 as configurable VLANs, you can assign them to different VLAN IDs

The **no** form of the command removes the VLAN.

## Examples

The following example shows how to create a port-based VLAN.

```
device(config)# vlan 222 by port
```

The following example shows the port-based VLAN configuration.

```
device(config)# vlan 10 name IP_VLAN by port
device(config-vlan-10)# untagged ethernet 1/1/1 to 1/1/6
added untagged port ethe 1/1/1 to 1/1/6 to port-vlan 10.
```

The following example shows how to create continuous and discontinuous VLANs.

```
device(config)# vlan 2 to 7 20 25
device(config-mvlan-2*25)#
```

The following example shows how to create continuous VLANs.

```
device(config)# vlan 2 to 7
device(config-mvlan-2-7)#
```

The following example shows how to create discontinuous VLANs.

```
device(config)# vlan 2 4 7
device(config-mvlan-2*7)#
```

# vlan-config

Configures Virtual Local Area Network (VLAN) tasks such as all or selective ports to a VLAN, moving untagged port membership between VLANs, and removing ports from a VLAN.

## Syntax

**vlan-config add** [**all-tagged** | **tagged-vlan** ]

**vlan-config move** [**untagged** *VLAN ID* ]

**vlan-config remove** [**all** | **vlan** *VLAN ID* ]

## Command Default

Active vlan preconfiguration is not a mandatory for **vlan-config add tagged-vlan** command. This command can create new VLAN even if it is not present. Maximum allowed discrete or set of VLAN(s) is 1024.

## Parameters

**add**
Adds a port to all the configured active VLANs.

**all-tagged**
Adds an interface to all VLANs as tagged members.

**tagged-vlan**
Adds an interface to selective VLANs as tagged members.

**move**
Moves an untagged port from one VLAN to another VLAN.

**untagged** *VLAN ID*
Moves the specific untagged VLAN port to another VLAN. It also moves the default VLAN of a dual mode port to another VLAN. The VLAN ID ranges from 1 to 4095.

**remove**
Removes a tagged or an untagged port from the VLAN.

**all**
Removes all VLANs from the physical port.

**vlan** *VLAN ID*
Removes the VLAN as specified by the VLAN ID from the physical port. The VLAN ID ranges from 1 to 4095.

## Modes

Interface configuration mode

# Usage Guidelines

Using the **vlan-config add** command, you can create a new VLAN and add the interface to it, if interface being added is the first interface. The command will also add port to non-active and non configured VLAN. It is not available on a private VLAN-enabled port and is not applicable to VLAN groups, MCT VLANs, GVRP, SPX PE ports, and flex-auth ports. The command is available in MIF mode. The maximum VLAN or VLAN range supported in a single input is 300.

**NOTE**
The command line prompt will not be available for the next command until the port is added to all VLANs in the system. The command is a non-savable command, which adds the interface as a tagged member. Command will not be available on a PVLAN Enabled port.

Using the **vlan-config move** command, you can move untagged ports from one VLAN to another without having to remove an untagged port from the old VLAN and to again add it to the new VLAN. This command can run on a multiple interface command mode.

**NOTE**

- If a new VLAN is not configured, the system allows creation of a new VLAN and the port is added to it. However, if the port is part of a port extender device and has allowed VLANs configured on it, then the system does not allow creation of a new VLAN.
- The VLAN port that is being moved should either be a dual mode port or should be part of a non-default VLAN. A port cannot be moved to or from a private VLAN.

# Examples

The following example adds an interface to all tagged VLANs in the system.

```
device(config)# interface ethernet 1/1/9
device(config-if-e1000-1/1/9) vlan-config add all-tagged
```

The following example adds an interface to selective VLANs in the system.

```
ddevice(config-if-e40000-1/1/1)#vlan-config add tagged-vlan
      DECIMAL   VLAN number
   <cr>
device(config-if-e40000-1/1/1)#vlan-config add tagged-vlan 101 102 103
INFO : Command may take approximately 0 Seconds
device(config-if-e40000-1/1/1)#
Port(s) ethe 1/1/1  add to 1 vlan(s) complete.....
device(config-if-e40000-1/1/1)#
```

The following example moves the specific untagged membership of 1/1/9 from a VLAN to VLAN 40 in the system.

```
device(config)# interface ethernet 1/1/9
device(config-if-e1000-1/1/9) vlan-config move untagged 40
```

The following example removes all VLANs from the physical port in the system.

```
device(config)# interface ethernet 1/1/9
device(config-if-e1000-1/1/9) vlan-config remove all
```

The following example removes selective VLANs from the physical port in the system.

```
device(config-if-e40000-1/1/1)#vlan-config remove vlan 107 108 109 110
device(config-if-e40000-1/1/1)#
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |
| 08.0.70 | This command was modified. |

# vlan-group

Configures a VLAN group.

## Syntax

**vlan-group** *num* **vlan** *vlan-id* [ **to** *vlan-id* ]

**no vlan-group** *num* **vlan** *vlan-id* [ **to** *vlan-id* ]

## Command Default

A VLAN group is not configured.

## Parameters

*num*

Specifies the group VLAN ID. The values can be from 1 through 32.

**vlan** *vlan-id*

Specifies the starting VLAN ID to create a VLAN group.

**to** *vlan-id*

Specifies the ending VLAN ID. This is a continuos range of individual VLAN IDs.

## Modes

Global configuration mode

## Usage Guidelines

Specify the low VLAN ID first and the high VLAN ID second. The command adds all of the specified VLANs to the VLAN group. You can add up to 256 VLANs with the command at one time.

If a VLAN within the range you specify is already configured, or if the range contains more than 256 VLANs, the VLAN group is not created and an error message is displayed.

To add more than 256 VLANs, enter the **add-vlan** command in VLAN group configuration mode.

To remove one or more VLANs, enter the **remove-vlan** command in VLAN group configuration mode.

The **no** form of the command deletes the VLAN group.

## Examples

The following example shows the VLAN group configuration.

```
device(config)# vlan-group 1 vlan 2 to 255
```

# voice-vlan

Creates a voice VLAN at the global level.

## Syntax

**voice-vlan** *vlan-id*

**no voice-vlan** *vlan-id*

## Command Default

A global voice VLAN is not configured.

## Parameters

*vlan-id*

Specifies the VLAN identifier. The range is from 1 through 4095 (excluding all reserved VLANs).

## Modes

Authentication configuration mode

## Usage Guidelines

The global voice VLAN is the default VLAN for voice traffic and is used:

- When the RADIUS server does not return VLAN information after authentication success.
- When the RADIUS server is not reachable for first authentication and **auth-timeout-action** is set to **success**.
- Any time that the RADIUS server is not reachable, **auth-timeout-action** is set to **critical vlan**, and **voice-vlan** is configured for critical action.
- When authentication fails, **auth-fail-action** is set to **restricted** and **voice-vlan** is configured for restricted action.

The **no** form of the command removes the global voice VLAN configuration.

## Examples

The following example shows how to configure VLAN 4 as the global voice VLAN.

```
device# configure terminal
device(config)# authentication
device(config-authen)# voice-vlan 4
```

## History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was introduced. |

# vrf

Configures a Virtual Routing and Forwarding (VRF) and enters VRF configuration mode.

## Syntax

**vrf** *vrf-name*

**no vrf** *vrf-name*

## Command Default

A VRF is not created.

## Parameters

*vrf-name*

Specifies the name of the VRF. The name can be up to 255 characters.

## Modes

Global configuration mode

## Usage Guidelines

ICX 7150 devices do not support VRFs.

The **no** form of the command removes the VRF.

## Examples

The following example configures a VRF and enters VRF configuration mode.

```
device(config)# vrf vrf1
device(config-vrf-vrf1)#
```

# vrf forwarding

Assigns a VRF routing instance to an interface.

## Syntax

**vrf forwarding** *vrf-name*

**no vrf forwarding** *vrf-name*

## Command Default

The default VRF.

## Parameters

*vrf-name*

Specifies the name of the VRF that the interface is being assigned to.

## Modes

Interface configuration mode

## Usage Guidelines

When the VRF is configured on a tunnel, all IPv4 and IPv6 addresses are removed. The tunnel loopback configuration is removed.

The **no** form of the command removes the VRF routing instance assigned to an interface. IP addresses and protocol configuration on this Layer 3 interface are removed.

## Examples

The following examples assigns a VRF instance to the Ethernet interface 1/1/1.

```
device(config)# interface ethernet 1/1/1
device(config-if-e1000-1/1/1)# vrf forwarding guest
```

The following example shows how to configure a forwarding VRF named red on an IPsec tunnel interface identified as 1.

```
device(config)# interface tunnel 1
device(config-tnif-1)# tunnel mode ipsec ipv4
device(config-tnif-1)# vrf forwarding red
```

# vsrp

Configures VSRP on a device.

## Syntax

**vsrp vrid** *vrid-num*

**no vsrp vrid** *vrid-num*

## Command Default

VSRP is not configured.

## Parameters

**vrid** *vrid-num*

Configures the VRID for the VLAN. The VRID range is from 1 through 255.

## Modes

VLAN configuration mode

## Usage Guidelines

The **no** form of the command clears the VSRP configuration.

## Examples

The following example shows how to configure the VRID.

```
device(config)# vlan 200
device(config-vlan-200)# vsrp vrid 1
```

# vsrp auth-type

Configures a simple text-string as a password in packets sent on the interface.

## Syntax

**vsrp auth-type** { **no-auth** | **simple-text-auth** *password* }

**no vsrp auth-type** { **no-auth** | **simple-text-auth** *password*}

## Command Default

By default, no authentication is configured.

## Parameters

**auth-type**
Configures the VSRP authentication type.

**no-auth**
Configures the VRID and interface without authentication.

**simple-text-auth** *password*
Configures the VRID to use simple text authentication with a password up to 8 characters long.

## Modes

VLAN configuration mode

## Usage Guidelines

If the interfaces on which you configure the VRID use authentication, the VSRP packets on those interfaces also must use the same authentication.

- No authentication - The interfaces do not use authentication.
- Simple - The interfaces use a simple text-string as a password in packets sent on the interface. If the interfaces use simple password authentication, the VRID configured on the interfaces must use the same authentication type and the same password.

## Examples

The following example shows how to configure a simple password.

```
device(config)# vlan 200
device(config-vlan-200)# vsrp auth-type simple-text-auth ourpword
```

# vsrp-aware

Configures the security features on a VSRP-aware device.

## Syntax

**vsrp-aware vrid** *vrid* **tc-vlan-flush**

**no vsrp-aware vrid** *vrid* **tc-vlan-flush**

**vsrp-aware vrid** *vrid* { **no-auth** | **simple-text-auth** *password* } { **port-list** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] } }

**no vsrp-aware vrid** *vrid* { **no-auth** | **simple-text-auth** *password* } { **port-list** { [ **ethernet** *unit/slot/port* [ **to** *unit/slot/port* ] ... ] [ **lag** *lag-id* [ **to** *lag-id* ] ... ] } }

## Command Default

VSRP-aware security features are not configured.

## Parameters

**vrid** *vrid*

Specifies the VRID of the VSRP device. The valid range is from 1 through 255.

**tc-vlan-flush**

Flushes the MAC addresses learned on the VSRP-aware VLAN upon topology change.

**no-auth**

Configures no authentication as the preferred VSRP-aware security method. The VSRP device will not accept incoming packets that have authentication strings.

**simple-text-auth** *password*

Defines an authentication string to accept incoming VSRP Hello packets. The password can be up to 8 characters in length.

**port-list**

Specifies the set of ports to include in the configuration.

**ethernet** *unit/slot/port* [ **to** *unit/slot/port* ]

Specifies the Ethernet ports, set of ports, or range of ports.

**lag** *lag-id* [ **to** *lag-id* ]

Specifies a LAG, set of LAGs, or range of LAGs to include in the port list.

**to**

Specifies a range of Ethernet interfaces or LAG IDs.

## Modes

VLAN configuration mode

# Usage Guidelines

When the **tc-vlan-flush** option is enabled, MAC addresses will be flushed at the VLAN level, instead of at the port level. MAC addresses will be flushed for every topology change received on the VSRP-aware ports. When you configure the **tc-vlan-flush** option on a VSRP-aware device, and the device receives VSRP Hello packets from the VSRP master, VSRP authentication is automatically configured. However, if the VSRP-aware device does not receive VSRP Hello packets from the VSRP master when the **tc-vlan-flush** option is configured, you must manually configure VSRP authentication.

You can specify a list of ports, separated by a space, or a range of ports, or you can combine lists and ranges.

You can combine individual Ethernet ports, Ethernet port ranges, LAGs, and LAG ranges in the same command if you wish.

The **no** form of the command clears the security features on the VSRP-aware device.

# Examples

The following example shows how to configure the MAC addresses to be flushed at the VLAN level.

```
device(config)# vlan 200
device(config-vlan-200)# vsrp-aware vrid 11 tc-vlan-flush
```

The following example shows how to configure a simple authentication string for the VSRP.

```
device(config)# vlan 10
device(config-vlan-10)# vsrp-aware vrid 3 simple-text-auth pri-key
```

The following example shows how to configure no authentication for the VSRP.

```
device(config)# vlan 10
device(config-vlan-10)# vsrp-aware vrid 2 no-auth
```

The following example shows how to configure no authentication for a range of Ethernet ports.

```
device(config)# vlan 10
device(config-vlan-10)# vsrp-aware vrid 4 no-auth port-list ethernet 1/1/1 to 1/1/4
```

# History

| Release version | Command history |
|---|---|
| 08.0.61 | This command was updated to include the LAG ID option. |

# web access-group

Configures an ACL that restricts web management access to the device.

## Syntax

**web access-group** { *acl-num* | *acl-name* | **ipv6** *ipv6-acl-name* }

**no web access-group** { *acl-num* | *acl-name* | **ipv6** *ipv6-acl-name* }

## Command Default

Web management access is not restricted.

## Parameters

*acl-num*
The standard access list number. The valid values are 1 through 99.

*acl-name*
The standard access list name.

**ipv6** *ipv6-acl-name*
The IPv6 access list name.

## Modes

Global configuration mode

## Usage Guidelines

The **no** form of the command removes the restriction of web management access for an ACL.

## Examples

The following example shows how to configure an ACL that restricts web management access to the device. In this example, ACL 12 is configured. The device denies web management access from the IP addresses listed in ACL 12 and permits web management access from all other IP addresses. Without the last ACL entry for permitting all packets, this ACL would deny web management access from all IP addresses.

```
device(config)# access-list 12 deny host 209.157.22.98 log
device(config)# access-list 12 deny 209.157.23.0 0.0.0.255 log
device(config)# access-list 12 deny 209.157.24.0/24 log
device(config)# access-list 12 permit any
device(config)# web access-group 12
device(config)# write memory
```

# web client

Restricts web management access to a host with a specified IP address.

## Syntax

**web client** {*ip-address* | **ipv6** *ipv6-address* }

**no web client** {*ip-address* | **ipv6** *ipv6-address* }

## Command Default

Web management access is not restricted.

## Parameters

*ip-address*
The IPv4 address of the host to which the web management access is restricted.

**ipv6** *ipv6-address*
The IPv6 address of the host to which the web management access is restricted.

## Modes

Global configuration mode

## Usage Guidelines

You can specify only one IP address with one command. However, you can enter the command ten times to specify up to ten IP addresses.

The **no** form of the command removes the web management access restriction.

## Examples

The following example shows how to restrict web management access to the host with IP address 192.168.10.1.

```
device(config)# web client 192.168.10.1
```

# web-management

Configures web management access options.

## Syntax

**web-management** [ **enable** { **vlan** *vlan-id* | **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ]... ] } ]

**no web-management** [ **enable** { **vlan** *vlan-id* | **ethernet** *unit/slot/port* [ **to** *unit/slot/port* | [ **ethernet** *unit/slot/port* **to** *unit/slot/port* | **ethernet** *unit/slot/port* ]... ] } ]

**web-management** [ **allow-no-password** | **connection-receive-timeout** *timeout-value* | **frame** { **bottom** | **front-panel** | **menu** } | **hp-top-tools** | **http** | **https** | **list-menu** | **page-menu** | **page-size** *size* | **session-timeout** *time* | **tcp-port** *port-num* ]

**no web-management** [ **allow-no-password** | **connection-receive-timeout** *timeout-value* | **frame** { **bottom** | **front-panel** | **menu** } | **hp-top-tools** | **http** | **https** | **list-menu** | **page-menu** | **page-size** *size* | **session-timeout** *time* | **tcp-port** *port-num* ]

**web-management** [ **refresh** { **front-panel** | **port-statistic** | **rmon** | **stp** | **tftp** } *refresh-time* ]

**no web-management** [ **refresh** { **front-panel** | **port-statistic** | **rmon** | **stp** | **tftp** } *refresh-time* ]

## Command Default

Web management is enabled.

## Parameters

**enable**

Enables web management only to clients in a specific VLAN or Ethernet interface.

**vlan** *vlan-id*

Specifies that web management should be enabled on the clients of the specified VLAN.

**ethernet** *unit/slot/port*

Specifies the Ethernet interface on which web management should be enabled.

**to** *unit/slot/port*

Specifies the range of Ethernet interfaces.

**allow-no-password**

Allows the web server to have no password.

**connection-receive-timeout** *timeout-value*

Specifies the web connection receive timeout.

**frame**

Enables a frame.

**bottom**

The bottom frame.

**front-panel**
The front-panel frame.

**menu**
The menu frame.

**hp-top-tools**
Enables the support of HP Top Tools.

**http**
Enables web management for HTTP access.

**https**
Enables web management for HTTPS access.

**list-menu**
Displays the web menu as a list.

**page-menu**
Enables the page menu.

**page-size** *size*
Configures the maximum number of entries on a page.

**session-timeout** *time*
Configures the web session timeout in seconds. Valid values are from 5 through 65000.

**tcp-port** *port-num*
Configures the HTTP port. The default port is 80.

**refresh**
Configures the page refresh (polling time) in seconds.

**front-panel**
Configures the front-panel refresh time.

**port-statistic**
Configures the port statistic refresh time.

**rmon**
Configures the RMON statistics refresh time.

**stp**
Configures the STP statistics refresh time.

**tftp**
Configures the TFTP statistics refresh time.

*refresh-time*
The refresh time in seconds.

# Modes

Global configuration mode

# Usage Guidelines

The **no** form of the command removes the web management configurations.

# Examples

The following example shows how to enable web management for HTTPS access.

```
device(config)# web-management https
```

The following example shows how to enable web management access only to clients connected to ports within port-based VLAN 10.

```
device(config)# web-management enable vlan 10
```

The following example shows how to enable web management access on a range of Ethernet interfaces.

```
device(config)# web-management enable ethernet 1/1/1 to 1/2/3
```

The following example shows how to configure the front-panel refresh time to 30 seconds.

```
device(config)# web-management refresh front-panel 30
```

# webauth

Configures a Web Authentication VLAN and enters the Web Authentication configuration mode.

## Syntax

**webauth**

**no webauth**

## Modes

VLAN configuration mode

## Usage Guidelines

Use the **enable** command in the Web Authentication configuration mode to enable Web Authentication.

Tthe **no** form of the command removes the Web Authentication VLAN.

## Examples

The following example shows how to configure a Web Authentication VLAN.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config(config-vlan-10-webauth)#
```

The following example deletes a Web Authentication VLAN.

```
device(config)# vlan 10
device(config-vlan-10)# no webauth
```

# webauth-redirect-address

Configures a redirect address for Web Authentication to prevent the display of error messages saying that the certificate does not match the name of the site.

## Syntax

**webauth-redirect-address** *address-string*

**no webauth-redirect-address** [ *address-string* ]

## Command Default

By default, the Web Authentication address returned to the browser is the IP address of the switch.

## Parameters

*address-string*

Specifies the redirect address. You can specify up to 64 alphanumeric characters.

## Modes

Global configuration mode

Web Authentication configuration mode

## Usage Guidelines

You can enter any value for the address string , but entering the name on the security certificate prevents the display of error messages saying that the security certificate does not match the name of the site.

On a Layer 2 device, the command is supported in Global configuration mode and on a Layer 3 device the command is supported in Web Authentication configuration mode.

The **no** form of the command resets the redirect address to that of the IP address of the switch.

## Examples

The following example shows how to set the Web Authentication redirect address on a Layer 3 switch.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webauth-redirect-address my.domain.net
```

# webpage custom-text

Customizes the text that appears on the title bar, login button, header, and footer on the Web Authentication pages.

## Syntax

**webpage custom-text** { **bottom** *footer* | **login-button** *button-text* | **title** *title-text* | **top** *header* }

**no webpage custom-text** { **bottom** *footer* | **login-button** *button-text* | **title** *title-text* | **top** *header* }

## Command Default

The default header text is "Welcome to Ruckus Networks Web Authentication Homepage".

The default title bar text is "Web Authentication".

The default login button text is "Login".

The default footer text is "This network is restricted to authorized users only. Violators may be subjected to legal prosecution. Activity on this network is monitored and may be used as evidence in a court of law. Copyright <*year*> Ruckus Networks."

## Parameters

**bottom** *footer*

Customizes the footer on a Web Authentication page. Specify up to 255 alphanumeric characters for the string.

**login-button** *button-text*

Customizes the login button that appears on the bottom of the Web Authentication Login page. Enter up to 32 alphanumeric characters for the string.

**title** *title-text*

Customizes the title bar that appears on all Web Authentication pages. You can specify up to 128 alphanumeric characters.

**top** *header*

Customizes the header that appears on all Web Authentication pages. You can specify up to 255 alphanumeric characters.

## Modes

Web Authentication configuration mode

## Usage Guidelines

You can use the **show webauth** command to view the configured text for Web Authentication pages.

The **no** form of the command resets the text to the default.

# Examples

The following example shows how to customize the text on the title bar.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage custom-text title "Ruckus Secure Access Page"
```

The following example shows how to customize the header that appears on all Web Authentication pages.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage custom-text top "Welcome to Network One"
```

The following example shows how to customize the login button that appears on the bottom of the Web Authentication Login page.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage custom-text login-button "Press to Log In"
```

The following example shows how to customize the footer that appears on all Web Authentication pages.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage custom-text bottom "Network One Copyright 2010"
```

# webpage logo

Customizes the logo that appears on all Web Authentication pages and its placement.

## Syntax

**webpage logo** { **copy tftp** { *ipv4-address* | *ipv6-address* } *file-name* | **align** { **left** | **center** | **right** } }

**no webpage logo** [ **copy tftp** { *ipv4-address* | *ipv6-address* } *file-name* | **align** [ **left** | **center** | **right** ] ]

## Command Default

By default, the logo is left-aligned at the top of the page.

## Parameters

**copy tftp**
Copies an image from the TFTP server to the switch.

*ipv4-address*
Specifies the IPv4 address of the TFTP server.

*ipv6-address*
Specifies the IPv6 address of the TFTP server.

*file-name*
Specifies the name of the file that must be copied from the TFTP server to the switch.

**align**
Configures the placement of the logo on the Web Authentication pages.

**left**
Aligns the logo to the left at the top of the page.

**right**
Aligns the logo to the right at the top of the page.

**center**
Aligns the logo to the center at the top of the page.

## Modes

Web Authentication configuration mode

## Usage Guidelines

To customize the banner image, use TFTP to upload an image file from a TFTP server to the FastIron switch. The image file can be in the jpg, bmp, or gif format, and its file size must be 64 KB or less. When you upload a new image file, it willl overwrite the existing image file.

The **no** form of the command deletes the logo from all Web Authentication pages and removes it from flash memory.

**NOTE**
The **webpage logo** command downloads the image file and stores it in the device flash memory. Therefore, it is not necessary to follow this command with a **write memory** command.

# Examples

The following example shows how to replace the existing logo with a new one.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage logo copy tftp 10.10.5.1 ruckuslogo.gif
```

The following example shows how to right-justify the log at the top of the page.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage logo align right
```

# webpage terms

Customizes the text box that appears on the Web Authentication Login page.

## Syntax

**webpage terms copy tftp** { *ipv4-address* | *ipv6-address* } *file-name*

**no webpage terms copy tftp** { *ipv4-address* | *ipv6-address* } *file-name*

## Command Default

By default, the text box is empty and is not visible.

## Parameters

**copy tftp**
Copies an ASCII text file from a TFTP server to the switch.

*ipv4-address*
The IPv4 address of the TFTP server.

*ipv6-address*
The IPv4 address of the TFTP server.

*file-name*
Specifies the name of the text file on the TFTP server.

## Modes

Web Authentication configuration mode

## Usage Guidelines

The text file size must not exceed 2 KB.

**NOTE**
The **webpage terms** command downloads the text file and stores it in the device flash memory. Therefore, it is not necessary to follow this command with a **write memory** command.

The **no** form of the command reverts back to the default; that is, the textbox is empty and not visible.

## Examples

The following example shows how to create or replace a text box.

```
device(config)# vlan 10
device(config-vlan-10)# webauth
device(config-vlan-10-webauth)# webpage terms copy tftp 10.10.5.1 policy.txt
```

# wpad

Specifies the Proxy Auto-Config (PAC) file location using the Web Proxy Auto-Discovery (WPAD) protocol.

## Syntax

**wpad** *ASCII -string*

**no wpad** *ASCII -string*

## Parameters

*ASCII-string*
The full network location of the PAC file.

## Modes

DHCP server pool configuration mode

## Usage Guidelines

The **no** form of the command removes the specified string from the server pool.

## Examples

The following example specifies the location of the PAC file.

```
device(config)# ip dhcp-server pool cabo
device(config-dhcp-cabo)# wpad http://172.26.67.243:8080/wpad.dat
```

## History

| Release version | Command history |
|---|---|
| 8.0.40 | This command was introduced. |

# write terminal

Displays the running configuration.

## Syntax

**write terminal**

## Modes

User EXEC mode

Privileged EXEC mode

Global configuration mode

## Usage Guidelines

This command performs the same function as the **show running-config** command.

# Examples

The following example displays the running configuration.

```
device(config)# write terminal
Current configuration:
!
ver 08.0.30
!
stack unit 1
  module 1 icx7450-24-port-management-module
  module 2 icx7400-xgf-4port-40g-module
  module 3 icx7400-qsfp-1port-40g-module
  module 4 icx7400-qsfp-1port-40g-module
!
!
!
vlan 1 name DEFAULT-VLAN by port
!
vlan 2 name IP_IPX_Protocol by port
!
vlan 10 by port
!
authentication
 disable-aging
!
boot sys fl sec
ip address 10.25.224.197 255.255.255.0 dynamic
ip dns domain-list englab.ruckuswireless.com
ip dns server-address 10.31.2.10
ip default-gateway 10.25.224.1
!
!
ntp
!
!
dot1x-mka-enable
!
!
sflow sample 566
sflow polling-interval 30
sflow max-packet-size 1200
sflow export cpu-traffic 18
sflow export system-info 30
sflow destination 2.2.2.2
sflow destination 3.3.3.3
sflow destination 4.4.4.4
sflow source-port 9999
sflow enable
!
!
end
```

# xwindow-manager

Specifies the IP addresses of systems that are running the X Window System Display Manager and are available to the client.

## Syntax

**xwindow-manager** *ip-address* [ *ip-address* ] [ *ip-address* ]

**no xwindow-manager** *ip-address* [*ip-address*] [*ip-address*]

## Parameters

*ip-address*

Specifies an IP address of the system running the X Window System Display Manager.

## Modes

DHCP server pool configuration mode

## Usage Guidelines

You can configure a maximum of three X Window System Display Manager IP addresses in a DHCP server pool.

The **no** form of the command removes the X Window System Display Manager IP addresses from the DHCP server pool.

## Examples

The following example configures the IP addresses of systems that are running the X Window System Display Manager in a DHCP server pool.

```
device(config)# ip dhcp-server pool cabo
device(config-dhcp-cabo)# xwindow-manager 10.38.12.1 10.38.12.3 10.38.12.5
```

## History

| Release version | Command history |
|---|---|
| 8.0.30b | This command was introduced. |

# zero-touch-enable

Allows the CB in a Campus Fabric (SPX) domain to discover PE candidates and convert them to active PE units.

## Syntax

**zero-touch-enable**

**no zero-touch-enable**

## Command Default

Disabled by default.

## Modes

CB configuration sub-mode.

## Usage Guidelines

The command is available only on an ICX 7750 or an ICX 7650 device configured as a CB.

The **no** form of the command disables zero-touch functions.

The command should be disabled if the user does not intend to discover new units in the domain.

Ruckus recommends removing **zero-touch-enable** configuration after all PEs are added.

The command cannot discover existing PE or provisional PE units.

You cannot use the **spx interactive-setup** or the **zero-touch-enable** command from a device running FastIron release 08.0.90 or later to discover PE units running a pre-08.0.90 release, due to a difference in message types beginning with the FastIron 08.0.90 release. Be sure to load the same FastIron 08.0.90 or later image to the potential PE units before executing either command.

Related commands:

- **zero-touch-ports**
- **spx interactive-setup**
- **spx zero-touch-deny**

## Examples

The following example enables the zero-touch utility. It also removes **spx zero-touch-deny** configuration, if present.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# zero-touch-enable
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |

# zero-touch-ports

Defines additional ports on which candidate PE units can be discovered when the zero touch provisioning utility or spx interactive-setup is enabled.

## Syntax

**zero-touch-ports** *portlist*

**no zero-touch-ports** *portlist*

## Command Default

By default, ports are not used for PE discovery.

## Parameters

*portlist*

Port, list of ports, port range, or combination to be used for discovering 802.1br (SPX) PE candidates.

## Modes

CB configuration mode.

## Usage Guidelines

The **no** form of the command disables zero-touch and spx-interactive probes on the specified ports and makes them available for other uses.

Only CB ports can be configured as zero-touch ports.

Port ranges for zero-touch ports are independent from SPX port or LAG ranges. Changing one range does not affect the other.

Ports designated as zero-touch ports are used only to discover new PE candidates. They do not modify existing SPX ports or LAGs. For example, if a user connects a new link between the CB and an existing PE unit, the new link is not discovered. The user must manually add or remove a port to or from an existing SPX link or SPX LAG.

Ruckus recommends that **zero-touch-ports** designation be removed once all candidate PEs have been discovered. Once the designation is removed, the ports can be configured for other purposes.

Related commands:

- **zero-touch-enable**
- **spx interactive-setup**

# Examples

The following example designates a range of ports from 1/1/10 through 1/1/20 as zero-touch-ports.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# zero-touch-ports 1/1/10 to 1/1/20
```

The following example designates three independent ports (2/1/5, 2/1/7, and 3/1/9) as well as a range of ports (3/1/2 to 3/1/5) as zero-touch ports.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# zero-touch-ports 2/1/5 2/1/7 3/1/2 to 3/1/5  3/1/9
```

The following example removes **zero-touch-ports** designation from two ports, 1/1/7 and 1/1/8.

```
device# configure terminal
device(config)# spx cb-configure
device(config-spx-cb)# no zero-touch-ports 1/1/7 to 1/1/8
```

# History

| Release version | Command history |
|---|---|
| 08.0.50 | This command was introduced. |



Ruckus Wireless, Inc., a wholly owned subsidiary of CommScope, Inc.
350 West Java Dr., Sunnyvale, CA 94089 USA
www.ruckuswireless.com