



# Ruckus Wireless™ ZoneDirector™

## Release 9.10 CLI Reference Guide

Part Number 800-70877-001 Rev A  
Published March 2015

[www.ruckuswireless.com](http://www.ruckuswireless.com)

## Copyright Notice and Proprietary Information

Copyright 2015. Ruckus Wireless, Inc. All rights reserved.

No part of this documentation may be used, reproduced, transmitted, or translated, in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without prior written permission of Ruckus Wireless, Inc. ("Ruckus"), or as expressly provided by under license from Ruckus.

### Destination Control Statement

Technical data contained in this publication may be subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

### Disclaimer

THIS DOCUMENTATION AND ALL INFORMATION CONTAINED HEREIN ("MATERIAL") IS PROVIDED FOR GENERAL INFORMATION PURPOSES ONLY. RUCKUS AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THE MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OR THAT THE MATERIAL IS ERROR-FREE, ACCURATE OR RELIABLE. RUCKUS RESERVES THE RIGHT TO MAKE CHANGES OR UPDATES TO THE MATERIAL AT ANY TIME.

### Limitation of Liability

IN NO EVENT SHALL RUCKUS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, OR DAMAGES FOR LOSS OF PROFITS, REVENUE, DATA OR USE, INCURRED BY YOU OR ANY THIRD PARTY, WHETHER IN AN ACTION IN CONTRACT OR TORT, ARISING FROM YOUR ACCESS TO, OR USE OF, THE MATERIAL.

### Trademarks

Ruckus Wireless, Ruckus, Bark Logo, BeamFlex, ChannelFly, Ruckus Pervasive Performance, SmartCell, ZoneFlex, Dynamic PSK, FlexMaster, MediaFlex, MetroFlex, Simply Better Wireless, SmartCast, SmartMesh, SmartSec, SpeedFlex, ZoneDirector, ZoneSwitch, and ZonePlanner are trademarks of Ruckus Wireless, Inc. in the United States and other countries. All other product or company names may be trademarks of their respective owners.

# Contents

## About This Guide

Document Conventions . . . . .	8
Documentation Feedback . . . . .	9

## 1 Understanding the ZoneDirector Command Line Interface

Introduction . . . . .	11
Accessing the Command Line Interface . . . . .	11
Requirements. . . . .	11
Step 1: Connecting the Administrative Computer to ZoneDirector . . . . .	11
Step 2: Start and Configure the SSH Client . . . . .	12
Step 3: Log Into the CLI . . . . .	16
Using the ? Command . . . . .	17
Top-Level Commands . . . . .	18
Using the Help Command . . . . .	19

## 2 Viewing Current Configuration

Show Commands Overview . . . . .	21
Show Location Services Commands. . . . .	21
Show AAA Commands . . . . .	22
Show DHCP Commands . . . . .	25
Show Access Point Commands . . . . .	26
Show AP Group Commands. . . . .	33
Show AP Policy Commands . . . . .	36
Show System Configuration Commands. . . . .	37
Show Performance Commands . . . . .	39
Show System Information Commands . . . . .	41
Show Ethernet Info Commands . . . . .	43
Show Technical Support Commands . . . . .	44
Show Management ACL Commands . . . . .	46
Show Static Route Commands. . . . .	47
Show WLAN Commands . . . . .	48
Show WLAN Group Commands . . . . .	51
Show L2 Access Control List Commands . . . . .	53
Show Whitelist Commands . . . . .	55

Show L3 Access Control List Commands . . . . .	57
Show Hotspot Commands . . . . .	59
Show Guest Policy Commands . . . . .	69
Show Hotspot 2.0 Operator Commands . . . . .	70
Show Hotspot 2.0 Service Provider Commands . . . . .	71
Show Role Commands . . . . .	71
Show VLAN Pool Commands . . . . .	73
Show User Commands . . . . .	74
Show Currently Active Clients Commands . . . . .	75
Show Mesh Commands . . . . .	78
Show Dynamic PSK Commands . . . . .	80
Show Dynamic Certificate Commands . . . . .	81
Show Guest Pass Commands . . . . .	81
Show Rogue Device Commands . . . . .	82
Show Events and Activities Commands . . . . .	83
Show Alarm Commands . . . . .	84
Show License Commands . . . . .	84
Show USB Software Commands . . . . .	85
Show Application Denial Policy Commands . . . . .	86
Show Session-Timeout Commands . . . . .	87
Show Active Wired Client Commands . . . . .	88
Show RADIUS Statistics Commands . . . . .	88
Show Load Balancing Commands . . . . .	90
Monitor AP MAC Commands . . . . .	91
Monitor Currently Active Client Commands . . . . .	93
Monitor Sysinfo Commands . . . . .	94

### 3 **Configuring Controller Settings**

Configuration Commands Overview . . . . .	98
General Config Commands . . . . .	98
Configure Context Show Commands . . . . .	99
Configure Location Services Commands . . . . .	102
Configure AAA Server Commands . . . . .	103
Configure DHCP Server Commands . . . . .	106
Configure Admin Commands . . . . .	109
Admin Authentication Commands . . . . .	110
Configure Access Points Commands . . . . .	112
Radio 2.4/5 GHz Commands . . . . .	120
AP Port Setting Commands . . . . .	134

Configure AP Policy Commands . . . . .	150
Configure AP Group Commands . . . . .	159
Configure Location Based Service Commands . . . . .	162
Radio 2.4/5 GHz Commands . . . . .	167
QoS Commands . . . . .	175
Model-Specific Commands . . . . .	176
AP Group Membership . . . . .	182
Model-Specific Port Settings . . . . .	184
LLDP Commands . . . . .	198
Configure Certificate Commands . . . . .	199
Configure Hotspot Redirect Settings . . . . .	201
Configure Layer 2 Access Control Commands . . . . .	202
Configure Layer 3 Access Control Commands . . . . .	210
Layer 3 IPv6 Access Control List Commands . . . . .	222
Configure Precedence Policy Commands . . . . .	225
Configure Precedence Policy Rule Commands . . . . .	225
Configure Device Policy Commands . . . . .	227
Configure Application Denial Policy Commands . . . . .	231
Configure Application Denial Policy Rules . . . . .	233
Configuring User-Defined Applications . . . . .	235
Configure Application Port Mapping . . . . .	237
Configure Whitelist Commands . . . . .	238
Configuring Whitelist Rules . . . . .	239
Configure Band Balancing Commands . . . . .	240
Configure Load Balancing Commands . . . . .	241
Configure STP Commands . . . . .	247
Configure System Commands . . . . .	247
Interface Commands . . . . .	249
Smart Redundancy Commands . . . . .	255
Management Interface Commands . . . . .	256
SNMPv2 Commands . . . . .	261
SNMPv3 Commands . . . . .	265
Syslog Settings Commands . . . . .	269
Management Access Control List Commands . . . . .	274
QoS Commands . . . . .	276
Management ACL Commands . . . . .	288
Configure UPNP Settings . . . . .	291
Configure Zero-IT Settings . . . . .	292
Configure Dynamic PSK Expiration . . . . .	293

Configure WLAN Settings Commands . . . . .	294
Configuring DHCP Option 82 Sub-Option Settings. . . . .	340
Configuring Dynamic PSKs . . . . .	347
Configure WLAN Group Settings Commands . . . . .	359
Configure Role Commands. . . . .	367
Configure VLAN Pool Commands . . . . .	379
Configure User Commands. . . . .	381
Configure Guest Access Commands . . . . .	387
Configuring Guest Access Restriction Rules. . . . .	395
IPv6 Guest Restrict Access Commands. . . . .	400
Configure Hotspot Commands . . . . .	407
Hotspot Access Restriction Commands. . . . .	427
Configure Hotspot 2.0 Commands . . . . .	433
Configure Mesh Commands . . . . .	453
Configure Alarm Commands. . . . .	460
Configure Alarm-Event Settings . . . . .	468
Configure Services Commands. . . . .	472
Configure WIPS Commands . . . . .	487
Configure Email Server Commands. . . . .	489
Configure SMS Server Commands . . . . .	491
Configure mDNS (Bonjour) Commands. . . . .	493
Configuring a Bonjour Policy . . . . .	494
Configuring mDNS Proxy Rules . . . . .	495

## 4 Using Debug Commands

Debug Commands Overview . . . . .	498
General Debug Commands . . . . .	498
Show Commands. . . . .	505
Accessing a Remote AP CLI . . . . .	510
Working with Debug Logs and Log Settings . . . . .	513
Remote Troubleshooting. . . . .	520
AP Core Dump Collection . . . . .	522
Script Execution . . . . .	524

## Index

# About This Guide

The *ZoneDirector Release 9.10 CLI Reference Guide* contains the syntax and commands for configuring and managing ZoneDirector from a command line interface.

This guide is written for service operators and system administrators who are responsible for managing, configuring, and troubleshooting Ruckus Wireless devices. Consequently, it assumes a basic working knowledge of local area networking, wireless networking, and wireless devices.

---

**NOTE** If release notes are shipped with your product and the information there differs from the information in this guide, follow the instructions in the release notes.

---

Most user guides and release notes are available in Adobe Acrobat Reader Portable Document Format (PDF) or HTML on the Ruckus Wireless Support Web site at <https://support.ruckuswireless.com/documents>.

# Document Conventions

The following two tables list the text and notice conventions that are used throughout this guide.

Table 1. Text conventions

<b>Convention</b>	<b>Description</b>	<b>Example</b>
monospace	Represents information as it appears on screen	[Device name]>
<b>monospace bold</b>	Represents information that you enter	[Device name]> <b>set ipaddr 10.0.0.12</b>
<b>default font bold</b>	Keyboard keys, software buttons, and field names	On the <b>Start</b> menu, click <b>All Programs</b> .
<i>italics</i>	Screen or page names	Click <b>Advanced Settings</b> . The <i>Advanced Settings</i> page appears.

Table 2. Notice conventions

<b>Notice Type</b>	<b>Description</b>
<b>NOTE</b>	Information that describes important features or instructions
<b>CAUTION!</b>	Information that alerts you to potential loss of data or potential damage to an application, system, or device
<b>WARNING!</b>	Information that alerts you to potential personal injury



## Documentation Feedback

Ruckus Wireless is interested in improving its documentation and welcomes your comments and suggestions. You can email your comments to Ruckus Wireless at:

[docs@ruckuswireless.com](mailto:docs@ruckuswireless.com)

When contacting us, please include the following information:

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

For example:

- ZoneDirector Release 9.10 CLI Reference Guide
- Part number: 800-70877-001 Revision A
- Page 88

# Understanding the ZoneDirector Command Line Interface

**1**

In this chapter:

- Introduction
- Accessing the Command Line Interface
- Using the Help Command
- Top-Level Commands

## Introduction

The Ruckus Wireless ZoneDirector Command Line Interface (CLI) is a software tool that enables you to configure and manage ZoneDirector, Ruckus Wireless's wireless LAN controller.

Using the command line interface, you can configure controller system settings, access points, wireless networks and client connection settings, or view current status information for each component of your ZoneFlex network. Each command performs a specific action for configuring device settings or returning information about the status of a specific device feature.

## Accessing the Command Line Interface

This section describes the requirements and the procedure for accessing the ZoneDirector CLI. The ZoneDirector CLI supports a maximum of 8 simultaneous SSH sessions, and maximum 4 sessions from the same IP address.

### Requirements

To access the ZoneDirector CLI, you will need the following:

- A computer that you want to designate as administrative computer
- A network connection to ZoneDirector, or
- An RS-232 serial cable (type depends on the ZoneDirector model):
  - If you are using ZoneDirector 3000/5000, you need an RS-232 serial to Ethernet cable.
  - If you are using ZoneDirector 1100, you need a DB-9 RS-232 to RS-232 cable.
- A Telnet or SSH (secure shell) client program

### Step 1: Connecting the Administrative Computer to ZoneDirector

The ZoneDirector Command Line Interface can be accessed in one of two ways:

- [Using an Ethernet Connection](#)
- [Using a Serial Connection](#)

## Using an Ethernet Connection

- 1 Ensure that ZoneDirector's IP address is reachable from the administrative computer. In factory default state, ZoneDirector's IP address is **192.168.0.2**.
- 2 Continue to ["Step 2: Start and Configure the SSH Client"](#).

## Using a Serial Connection

The steps for connecting the administrative computer directly to ZoneDirector using a serial cable depend on the ZoneDirector model that you are using. Refer to the relevant section below.

- [Connecting ZoneDirector 1100](#)
- [Connecting ZoneDirector 1200/3000/5000](#)

---

**NOTE** Before continuing, make sure that both the administrative computer and ZoneDirector are both powered on.

---

### ***Connecting ZoneDirector 1100***

For ZoneDirector 1100, you need a DB-9 RS-232 to RS-232 cable.

- 1 Connect one end of the RS-232 cable to the port labeled *Console* on ZoneDirector.
- 2 Connect the other end of the RS-232 cable to a COM port on the administrative computer.

### ***Connecting ZoneDirector 1200/3000/5000***

For ZoneDirector 1200/3000/5000, you need an RS-232 to Ethernet cable.

- 1 Connect the RJ-45 end of the cable to the port labeled *Console* on ZoneDirector.
- 2 Connect the RS-232 end of the cable to a COM port on the administrative computer.

## Step 2: Start and Configure the SSH Client

Before starting this procedure, make sure that your SSH client is already installed on the administrative computer.

---

**NOTE** The following procedure uses PuTTY, a free and open source Telnet/SSH client, for accessing the ZoneDirector CLI. If you are using a different Telnet/SSH client, the procedure may be slightly different (although the connection settings should be the same). For more information on PuTTY, visit [www.putty.org](http://www.putty.org).

---

## Using SSH

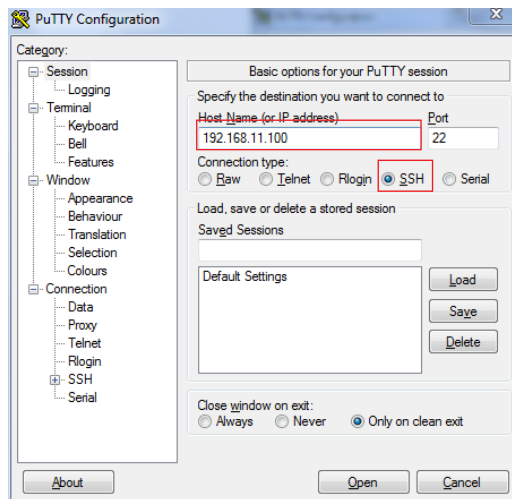
To start and configure the SSH client

- 1 Start PuTTY. The PuTTY Configuration dialog box appears, showing the *Session* screen.
  - 2 In *Connection type*, select **SSH**.
- 

**NOTE** Telnet access is disabled by default for security reasons. SSH is the recommended access method and you will not be allowed to access the ZoneDirector CLI via Telnet unless you have specifically enabled Telnet access.

---

Figure 1. Selecting SSH as the connection type



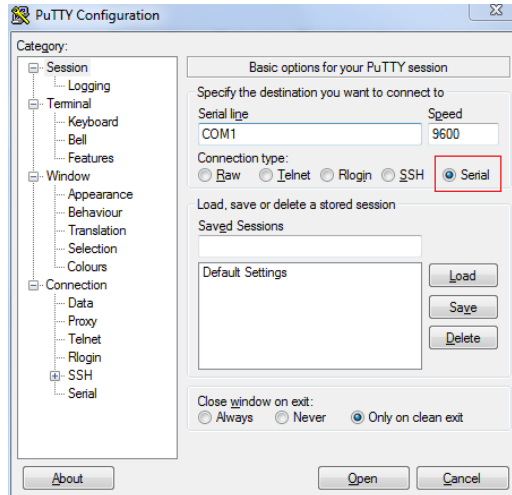
- 3 Enter the ZoneDirector IP address in the **Host Name (or IP address)** field.
- 4 Click **Open**. The PuTTY console appears and displays the login prompt.

## Using a Serial Connection

To start and configure the SSH client:

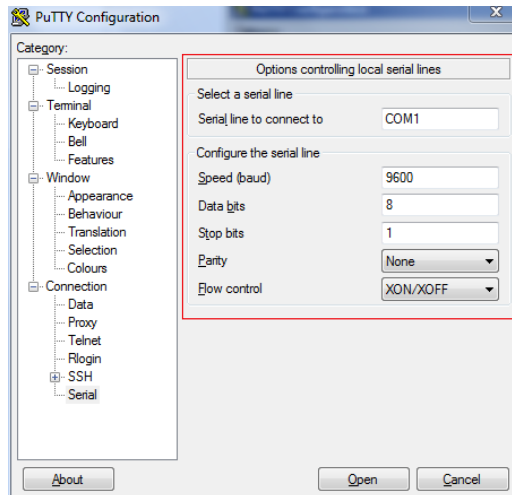
- 1 Start PuTTY. The PuTTY Configuration dialog box appears, showing the *Session* screen.
- 2 In *Connection type*, select **Serial** if you are connecting via serial cable.

Figure 2. Select Serial as the connection type



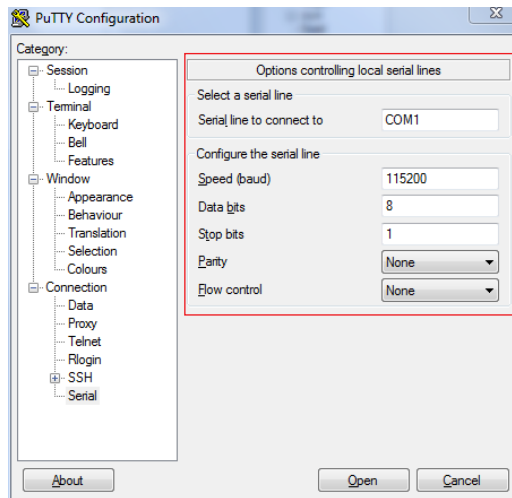
- 3 Under *Category*, click **Connection > Serial**. The serial connection options appear on the right side of the dialog box, displaying PuTTY's default serial connection settings.

Figure 3. PuTTY's default serial connection settings



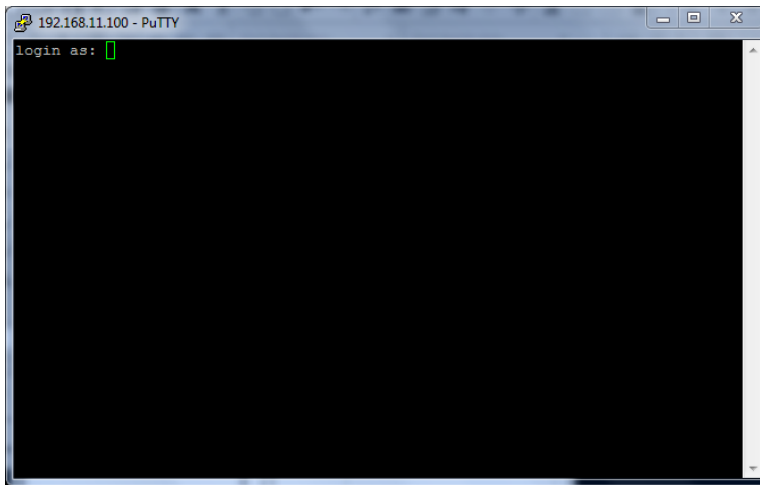
- 4 Configure the serial connection settings as follows:
  - *Serial line to connect to:* Type the COM port name to which you connected the RS-232 cable.
  - *Bits per second:* 115200
  - *Data bits:* 8
  - *Stop bits:* 1
  - *Parity:* None
  - *Flow control:* None

Figure 4. PuTTY's serial connection settings for connecting to ZoneDirector



- 5 Click **Open**. The PuTTY console appears and displays the login prompt.

Figure 5. The PuTTY console displaying the login prompt



You have completed configuring the Telnet/SSH client to connect to ZoneDirector.

### Step 3: Log Into the CLI

- 1 At the `login as` prompt, press <Enter> once.
- 1 At the `Please login` prompt, enter the ZoneDirector login name (default: **admin**), and then press <Enter>.
- 2 At the `Password` prompt, enter the ZoneDirector login password (default: **admin**), and then press <Enter>. The Ruckus Wireless ZoneDirector CLI welcome message and the `ruckus>` prompt appears.

You are now logged into the ZoneDirector CLI as a user with limited privileges. As a user with limited privileges, you can view a history of commands that were previously executed and ping a device. If you want to run more commands, you can switch to privileged mode by entering **enable** at the root prompt.

To view a list of commands that are available at the root level, enter **help** or?

---

**NOTE** You can tell if you are logged into the CLI in limited or privileged mode by looking at the `ruckus` prompt. If you are in limited mode, the prompt appears as `ruckus>` (with a *greater than* sign). If you are in privileged mode, the prompt appears as `ruckus#` (with a pound sign).

---



---

**NOTE** To enable privileged mode when another user session is enabled, use the <force> option with the enable command to force disconnect of the previous user session. (i.e., **enable force**).

---

## Using the ? Command

To display a brief list of commands that are available within a specific context, use the ? command.

### Example

To display commands within the debug context, enter the following command:

```
ruckus# debug
ruckus(debug) # ?
```

help	Shows available commands.
list-all	Lists all available commands.
history	Shows a list of previously run commands.
quit	Exits the debug context.
fw_upgrade	Upgrades the controller's firmware.
delete-station <MAC>	Disassociates a station.
restart-ap <MAC>	Restarts a device.
wlaninfo	Configures and enables debugging of WLAN service settings.
show	Contains commands that can be executed from within the context.
ps	Displays information about all processes that are running (ps -aux).
save debug_info <IP-ADDR> <FILE-NAME>	Saves debug information.
remote_ap_cli	Executes AP CLI command in remote AP.
save-config <IP-ADDR> <FILE-NAME>	Upload the configuration to the designated TFTP site.

logs	Contains commands that can be executed from within the context.
no	Contains commands that can be executed from within the context.
remote-troubleshooting	Troubleshooting commands group.
collect_ap_coredump	Enable AP core dump collection.
script	Manages system script for debug.

## Top-Level Commands

The following table lists the top-level CLI commands available in privileged mode.

exit	End the CLI session.
help	Show available commands.
quit	End the CLI session.
history	Show a list of previously run commands.
disable	Disable privileged commands.
ping <IP-ADDR/ DOMAIN-NAME>	Send ICMP echo packets to an IP/IPv6 address or domain name.
reboot	Reboot the controller.
shutdown	Shut down ZoneDirector, to power on ZoneDirector again, press the power.
set-factory	Reset the controller to factory defaults.
config	Enter the config context.
logo	Configure Ruckus logo. Options are “logo nodog” and “logo default.”
debug	Enter the debug context.
show	Display system options and settings.
reset	Reset RADIUS statistics commands.

---

<code>session-timeout</code> <NUMBER>	Set the CLI session timeout.
<code>monitor</code>	Begin system status monitoring.

---

## Using the Help Command

To display all commands that the Ruckus Wireless CLI supports, use the `help` command.

---

**NOTE** Entering the `help` command into the CLI prints a long list of commands on the screen. If you only want to view the commands that are available from within a specific context, use the `?` command. See [Using the ? Command](#) above for more information.

---

# Viewing Current Configuration

## 2

In this chapter:

- Show Commands Overview
- Show Location Services Commands
- Show AAA Commands
- Show DHCP Commands
- Show Access Point Commands
- Show AP Group Commands
- Show System Configuration Commands
- Show System Information Commands
- Show WLAN Commands
- Show Hotspot Commands
- Show Guest Policy Commands
- Show User Commands
- Show Mesh Commands
- Show Guest Pass Commands
- Show Events and Activities Commands
- Show Alarm Commands
- Monitor Sysinfo Commands

## Show Commands Overview

Show commands display the controller's current configuration and status information, such as system status and system configuration settings, along with the status and configurations of the controller's WLAN services, users, roles, AAA servers, access points, connected clients, AP groups and WLAN groups, etc.

Monitor commands allow the administrator to enter monitoring mode to view status and configuration changes as they occur.

## Show Location Services Commands

Use the `show location-services` commands to display information about the location servers that have been configured on the controller.

### **show location-services all**

To display a list of all location services servers that have been added to the controller, use the following command:

```
show location-services all
```

### **Syntax Description**

<code>show</code>	Display information
<code>location-services</code>	Display location server information
<code>all</code>	All location servers

### **Defaults**

None.

### **Example**

```
ruckus# show location-services all
Venue:
  ID:
  1:
    Status                = Disabled
    Venue Name            = MyVenue
```

```
Location Server FQDN = lbls.ruckuslbs.com
Location Server Port = 8883
Location Server PSK = password
```

```
ruckus#
```

## show location-services name

To display information on the specified location server, use the following command:

```
show location-services name <WORD>
```

## Show AAA Commands

Use the `show aaa` commands to display information about the authentication, authorization and accounting servers (AAA) servers that have been added to the controller.

### show aaa all

To display a list of all AAA servers that have been added to the controller, use the following command:

```
show aaa all
```

### Syntax Description

show	Display information
aaa	Display AAA server information
all	All AAA servers

### Defaults

None.

### Example

```
ruckus# show aaa all
AAA:
ID:
```

```
1:
Name= Local Database
Type= Local

2:
Name= Guest Accounts
Type= Guest

3:
Name= RADIUS Accounting
Type= RADIUS Accounting server
Primary RADIUS Accounting:
IP Address= 192.168.11.7
Port= 1813
Secret= secret
Secondary RADIUS Accounting:
Status= Disabled

4:
Name= Ruckus RADIUS
Type= RADIUS server
Auth Method=
Primary RADIUS:
IP Address= 192.168.11.99
Port= 1812
Secret= secret
Secondary RADIUS:
Status= Disabled

5:
Name= Ruckus AD
Type= Active Directory
IP Address= 192.168.11.17
Port= 389
Windows Domain Name= domain.ruckuswireless.com
Global Catalog= Disabled
Admin DN=domain
Admin Password=password
```

```
ruckus#
```

## show aaa name

To display information about a specific AAA server that has been added to the controller, use the following command:

```
show aaa name <WORD>
```

### Syntax Description

show	Display information
aaa name	Display information about the specified AAA server name
<WORD>	Name of the AAA server

### Defaults

None.

### Example

```
ruckus# show aaa name "Ruckus RADIUS"
```

```
AAA:
```

```
ID:
```

```
4:
```

```
Name= Ruckus RADIUS
```

```
Type= RADIUS server
```

```
Auth Method=
```

```
Primary RADIUS:
```

```
IP Address= 192.168.11.99
```

```
Port= 1812
```

```
Secret= secret
```

```
Secondary RADIUS:
```

```
Status= Disabled
```

```
ruckus#
```



## Show DHCP Commands

Use the `show dhcp` commands to display the current settings for any DHCP servers configured for DHCP relay agent use.

### **show dhcp all**

To display a list of all DHCP servers that have been configured on the controller, use the following command:

```
show dhcp all
```

### **Syntax Description**

<code>show</code>	Display information
<code>dhcp</code>	Display information about the specified DHCP server name
<code>all</code>	Display a list of all DHCP servers

### **Defaults**

None.

### **Example**

```
ruckus# show dhcp all
DHCP servers for DHCP relay agent:
ID:
  1:
    Name= DHCP Server 1
    Description=
    IP Address= 192.168.11.1
    IP Address=
```

```
ruckus#
```

### **show dhcp name**

To display a list of all DHCP servers that have been configured on the controller, use the following command:

```
show dhcp name <WORD>
```

## Syntax Description

show	Display information
dhcp	Display information about the specified DHCP server name
name	Display the DHCP server specified
<WORD>	Name of the DHCP server

## Defaults

None.

## Example

```
ruckus# show dhcp name "DHCP Server 1"
```

```
DHCP servers for DHCP relay agent:
```

```
  ID:
```

```
  1:
```

```
    Name= DHCP Server 1
```

```
    Description=
```

```
    IP Address= 192.168.11.1
```

```
    IP Address=
```

```
ruckus#
```

# Show Access Point Commands

Use the `show ap` commands to display the current settings of managed devices, including their network address settings, device names, radio settings, and others.

## show ap all

To display a summary of all devices that have been approved, use the following command:

```
show ap all
```

## Syntax Description

show	Display information
------	---------------------

---

ap	Show device information
all	All devices that have been approved by the controller

---

### **Defaults**

None.

### **Example**

```
ruckus# show ap all
AP:
ID:
1:
MAC Address= 04:4f:aa:0c:b1:00
Model= zf7962
Approved= Yes
Device Name= 7962 - MAP
Description= 7962 MAP (Living Room)
Location= Living Room
GPS=
Group Name= System Default
Radio a/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Radio b/g/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Override global ap-model port configuration= No
Network Setting:
Protocol mode= IPv4 and IPv6
Device IP Settings= Keep AP's Setting
IP Type= DHCP
```

```
IP Address= 192.168.11.6
Netmask= 255.255.255.0
Gateway= 192.168.11.1
Primary DNS Server=
Secondary DNS Server=

Device IPv6 Settings= Keep AP's Setting
IPv6 Type= Auto Configuration
IPv6 Address= fc00::1
IPv6 Prefix Length= 7
IPv6 Gateway=
IPv6 Primary DNS Server=
IPv6 Secondary DNS Server=
Mesh:
Status= Enabled
Mode= Auto
Uplink:
Status= Smart

2:
MAC Address= 00:24:82:3f:14:60
Model= zf7363
Approved= Yes
Device Name= 7363 - RAP
Description= 7363 - RAP (Study)
Location= Study
GPS=
Group Name= System Default
Radio a/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Radio b/g/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
```

```
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Override global ap-model port configuration= No
Network Setting:
Protocol mode= IPv4 and IPv6
Device IP Settings= Keep AP's Setting
IP Type= DHCP
IP Address= 192.168.11.3
Netmask= 255.255.255.0
Gateway= 192.168.11.1
Primary DNS Server= 192.168.11.1
Secondary DNS Server=

Device IPv6 Settings= Keep AP's Setting
IPv6 Type= Auto Configuration
IPv6 Address=
IPv6 Prefix Length= 7
IPv6 Gateway=
IPv6 Primary DNS Server=
IPv6 Secondary DNS Server=
Mesh:
Status= Enabled
Mode= Auto
Uplink:
Status= Smart
```

```
ruckus#
```

## **show ap devname**

To display information about a specific device using its device name, use the following command:

```
show ap devname <WORD>
```

### ***Syntax Description***

---

show	Display information
------	---------------------

---

---

ap devname	Show information about the specified device name
<WORD>	The name of the device

---

### **Defaults**

None.

### **Example**

```
ruckus# show ap devname "7962 - MAP"
AP:
ID:
1:
MAC Address= 04:4f:aa:0c:b1:00
Model= zf7962
Approved= Yes
Device Name= 7962 - MAP
Description= 7962 MAP (Living Room)
Location= Living Room
GPS=
Group Name= System Default
Radio a/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Radio b/g/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Override global ap-model port configuration= No
Network Setting:
Protocol mode= IPv4 and IPv6
Device IP Settings= Keep AP's Setting
IP Type= DHCP
```

```

IP Address= 192.168.11.6
Netmask= 255.255.255.0
Gateway= 192.168.11.1
Primary DNS Server=
Secondary DNS Server=

Device IPv6 Settings= Keep AP's Setting
IPv6 Type= Auto Configuration
IPv6 Address= fc00::1
IPv6 Prefix Length= 7
IPv6 Gateway=
IPv6 Primary DNS Server=
IPv6 Secondary DNS Server=
Mesh:
Status= Enabled
Mode= Auto
Uplink:
Status= Smart

ruckus#

```

## show ap mac

To search for the device that matches the specified MAC address, use the following command:

```
show ap mac <MAC>
```

### ***Syntax Description***

show	Display information
ap mac	Display information about the device with the specified MAC address
<MAC>	The MAC address of the device

### ***Defaults***

None.

**Example**

```
ruckus# show ap mac 04:4f:aa:0c:b1:00
AP:
ID:
1:
MAC Address= 04:4f:aa:0c:b1:00
Model= zf7962
Approved= Yes
Device Name= 7962 - MAP
Description= 7962 MAP (Living Room)
Location= Living Room
GPS=
Group Name= System Default
Radio a/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Radio b/g/n:
Channelization= Auto
Channel= Auto
WLAN Services enabled= Yes
5.8GHz Channels = Disabled
Tx. Power= Auto
WLAN Group Name= Default
Override global ap-model port configuration= No
Network Setting:
Protocol mode= IPv4 and IPv6
Device IP Settings= Keep AP's Setting
IP Type= DHCP
IP Address= 192.168.11.6
Netmask= 255.255.255.0
Gateway= 192.168.11.1
Primary DNS Server=
Secondary DNS Server=
```



```

Device IPv6 Settings= Keep AP's Setting
IPv6 Type= Auto Configuration
IPv6 Address= fc00::1
IPv6 Prefix Length= 7
IPv6 Gateway=
IPv6 Primary DNS Server=
IPv6 Secondary DNS Server=
Mesh:
Status= Enabled
Mode= Auto
Uplink:
Status= Smart

ruckus#

```

## Show AP Group Commands

Use the show ap-group commands to display Access Point Group settings.

### show ap-group all

To display all AP groups and their settings (including the default AP group), use the following command:

```
show ap-group all
```

### Syntax Description

show	Display information
ap-group	Display access point group information
all	All AP groups

### Defaults

None.

### Example

```

ruckus# show ap-group all
APGROUP:
ID:

```

```
1:
Name= System Default
Description= System default group for Access Points
Radio 11bgn:
Channelization= Auto
Channel= Auto
Enable auto channel selection which select from 1,6,11= Yes
Tx. Power= Auto
11N only Mode= Auto
WLAN Group= Default
Radio 11an:
Channelization= Auto
Channel= Auto
Tx. Power= Auto
11N only Mode= Auto
WLAN Group= Default
Members:
MAC= 04:4f:aa:0c:b1:00
MAC= 00:24:82:3f:14:60
MAC= 74:91:1a:2b:ff:a0
```

## APGROUP:

```
ID:
2:
Name= ap group 2
Description=
Radio 11bgn:
Channelization= Auto
Channel= Auto
Enable auto channel selection which select from 1,6,11= Yes
Tx. Power= Auto
11N only Mode= Auto
WLAN Group= Default
Radio 11an:
Channelization= Auto
Channel= Auto
Tx. Power= Auto
11N only Mode= Auto
WLAN Group= Default
```

```

Members:

APGROUP:
  ID:
  3:
  Name= ap group 1
  Description=
  Radio 11bgn:
  Channelization= Auto
  Channel= Auto
  Enable auto channel selection which select from 1,6,11= Yes
  Tx. Power= Auto
  11N only Mode= Auto
  WLAN Group= Default
  Radio 11an:
  Channelization= Auto
  Channel= Auto
  Tx. Power= Auto
  11N only Mode= Auto
  WLAN Group= Default
  Members:

ruckus#

```

## show ap-group name

To display details about a specific AP group, use the following command:

```
show ap-group name <WORD>
```

### **Syntax Description**

show	Display information
ap-group name	Display information about the AP group with the specified name
<WORD>	The name of the AP group

### **Defaults**

None.

**Example**

```
ruckus# show ap-group name "System Default"
APGROUP:
  ID:
  1:
  Name= System Default
  Description= System default group for Access Points
  Radio 11bgn:
  Channelization= Auto
  Channel= Auto
  Enable auto channel selection which select from 1,6,11= Yes
  Tx. Power= Auto
  11N only Mode= Auto
  WLAN Group= Default
  Radio 11an:
  Channelization= Auto
  Channel= Auto
  Tx. Power= Auto
  11N only Mode= Auto
  WLAN Group= Default
  Members:
  MAC= 04:4f:aa:0c:b1:00
  MAC= 00:24:82:3f:14:60
  MAC= 74:91:1a:2b:ff:a0

ruckus#
```

## Show AP Policy Commands

Use the show ap-policy command to display global access point policies that have been configured on the controller.

**show ap-policy**

```
show ap-policy
```

**Example**

```
ruckus# show ap-policy
```

```
Automatically approve all join requests from APs= Enabled
Limited ZD Discovery:
Status= Disabled
Management VLAN:
Status= Keep AP's setting
Balances the number of clients across adjacent APs= Disabled
Max. clients for 11BG radio= 100
Max. clients for 11N radio= 100
LWAPP message MTU= 1450
ruckus#
```

## Show System Configuration Commands

Use the `show config` commands to display the controller's system configuration settings.

### show config

To display the current system configuration settings, including network addressing, management VLAN, country code, logging, AAA servers, WLAN services, WLAN groups, AP list, SNMP, and ACLs, etc., use the following command:

```
show config
```

### Syntax Description

<code>show</code>	Display information
<code>config</code>	Display system configuration settings

### Defaults

None.

### Example

```
ruckus# show config
Protocol Mode= IPv4-Only
Device IP Address:
Mode= Manual
```

IP Address= 192.168.40.100  
Netmask= 255.255.255.0  
Gateway Address= 192.168.40.1  
Primary DNS= 192.168.40.1  
Secondary DNS=

Management VLAN:  
VLAN ID= 1

Country Code:  
Code= United States

Identity:  
Name= ZoneDirector

NTP:  
Status= Enabled  
Address= ntp.ruckuswireless.com

Log:  
Status= Disabled  
Address= 192.168.3.10  
Facility= local0  
Priority= emerg  
AP Facility= local0  
AP Priority= emerg

Tunnel MTU:  
Tunnel MTU= 1500

Bonjour Service:  
Status= Disabled

Telnet Server:  
Status= Disabled

FTP Server:  
Status= Enabled  
Anonymous Status= Enabled

```
FlexMaster:
  Status= Disabled
  Address=
  Interval= 15

AAA:
  ID:
    1:
      Name= Local Database
      Type= Local

    2:
      Name= Guest Accounts
      Type= Guest

  ...
  ...
ruckus#
```

## Show Performance Commands

Use the show performance commands to display performance details on an AP radio or client station.

### show performance

Use the following command to display performance details:

```
show performance
```

### show performance ap-radio2-4

Use the following command to display performance details for the AP's 2.4 GHz radio.

```
show performance ap-radio2-4 mac <MAC>
```

### Syntax Description

---

show performance	Display performance information
ap-radio-2-4	Display AP 2.4 GHz radio performance

---

---

mac <MAC>	The MAC address of the AP
-----------	---------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus# show performance ap-radio2-4 mac c4:10:8a:1f:d1:f0
AP performance:
  1:
    Radio b/g/n:
    MAC Address= c4:10:8a:1f:d1:f0
    Estimated Capacity= 9930
    Downlink= 67
    Uplink= 0
    RF pollution= 11
    Associated clients= 1
    Other APs= 0
ruckus#
```

### **show performance ap-radio5**

Use the following command to display performance details for the AP's 5 GHz radio:

```
show performance ap-radio5 mac <MAC>
```

### **Syntax Description**

---

show performance	Display performance information
ap-radio-5	Display AP 5 GHz radio performance
mac <MAC>	The MAC address of the AP

---

### **Defaults**

None.

### **Example**

```
ruckus# show performance ap-radio5 mac c4:10:8a:1f:d1:f0
AP performance:
  1:
```



```
Radio a/n:  
MAC Address= c4:10:8a:1f:d1:f0  
Estimated Capacity= 20891  
Downlink= 77  
Uplink= 2  
RF pollution= 3  
Associated clients= 1  
Other APs= 0
```

```
ruckus#
```

## **show performance station**

Use the following command to display performance details for a connected client/station:

```
show performance station mac <MAC>
```

### ***Syntax Description***

show performance	Display performance information
station	Display station performance
mac <MAC>	The MAC address of the station

### ***Defaults***

None.

### ***Example***

```
ruckus# show performance station mac 00:22:fb:ad:1b:2e  
Station performance:  
    MAC Address= 00:22:fb:ad:1b:2e  
    Estimated Capacity= 61401  
    Downlink= 76  
    Uplink= 18  
ruckus#
```

## **Show System Information Commands**

Use the `show sysinfo` commands to display the controller's system information.

## show sysinfo

To display an overview of the system status, including system, devices, usage summary, user activities, system activities, used access points, and support information, use the following command:

```
show sysinfo
```

### Syntax Description

---

show	Display information
sysinfo	Display an overview of various system statuses

---

### Defaults

None.

### Example

```
ruckus# show sysinfo
System Overview:
  Name= ZoneDirector
  IP Address= 192.168.40.100
  MAC Address= 00:13:11:01:01:01
  Uptime= 4d 0h 18m
  Model= ZD1112
  Licensed APs= 12
  Serial Number= 000000000011
  Version= 9.8.0.0 build 112
```

```
Devices Overview:
  Number of APs= 3
  Number of Client Devices= 2
  Number of Rogue Devices= 15
```

```
Usage Summary:
  Usage of 1 hr:
    Max. Concurrent Users= 2
    Bytes Transmitted= 45.87M
    Number of Rogue Devices= 15
  Usage of 24 hr:
    Max. Concurrent Users= 3
```

```
Bytes Transmitted= 5.90G
Number of Rogue Devices= 50
```

```
Memory Utilization:
Used Bytes= 61009920
Used Percentage= 47%
Free Bytes= 67158016
Free Percentage= 53%
```

```
ruckus#
```

## Show Ethernet Info Commands

Use the show ethinfo command to display current system Ethernet status.

### show ethinfo

```
show ethinfo
```

### Syntax Description

---

show	Display information
ethinfo	Display the current system Ethernet status

---

### Defaults

None.

### Example

```
ruckus# show ethinfo
System Ethernet Overview:
Port 0:
  Interface= eth0
  MAC Address= 00:13:11:01:01:01
  Physical Link= up
  Speed= 1000Mbps
Port 1:
  Interface= eth1
  MAC Address= 00:13:11:01:01:02
```

```
Physical Link= up  
Speed= 100Mbps
```

```
ruckus#
```

## Show Technical Support Commands

Use the following commands to display information that Ruckus Wireless may need when providing technical support.

### show techsupport

To display system information required by Technical Support, use the following command:

```
show techsupport
```

### Syntax Description

---

show	Display information
techsupport	Display information about the controller that may be required by Ruckus Wireless Technical Support

---

### Defaults

None.

### Example

```
ruckus# show techsupport  
ruckus# show techsupport  
System Overview:  
  Name= ZoneDirector  
  IP Address= 192.168.40.100  
  MAC Address= 00:13:11:01:01:01  
  Uptime= 15d 18h 44m  
  Model= ZD1112  
  Licensed APs= 12  
  Serial Number= 000000000011  
  Version= 9.7.0.0 build 155
```

Devices Overview:

Number of APs= 3  
Number of Client Devices= 2  
Number of Rogue Devices= 0

Usage Summary:

Usage of 1 hr:  
Max. Concurrent Users= 2  
Bytes Transmitted= 76.66M  
Number of Rogue Devices= 0  
Usage of 24 hr:  
Max. Concurrent Users= 0  
Bytes Transmitted= 2.24G  
Number of Rogue Devices= 0

Memory Utilization:

Used Bytes= 95956992  
Used Percentage= 74%  
Free Bytes= 32210944  
Free Percentage= 26%

Protocol Mode= IPv4-Only

Device IP Address:

Mode= Manual  
IP Address= 192.168.40.100  
Netmask= 255.255.255.0  
Gateway Address= 192.168.40.1  
Primary DNS= 192.168.40.1  
Secondary DNS=

Management VLAN:

VLAN ID= 1

Country Code:

Code= United States

Identity:

Name= ZoneDirector

...

```
...  
ruckus#
```

## Show Management ACL Commands

Use the `mgmt-acl` and `mgmt-acl-ipv6` commands to display information about the management access control lists configured on the controller.

### show mgmt-acl all

To display all management ACLs that have been configured on the controller, use the following command:

```
show mgmt-acl all
```

### show mgmt-acl name

To display information about a specific management ACL, use the following command:

```
show mgmt-acl name <NAME>
```

### show mgmt-acl-ipv6 all

To display all management ACLs that have been configured on the controller, use the following command:

```
show mgmt-acl-ipv6 all
```

### show mgmt-acl-ipv6 name

To display information about a specific management ACL, use the following command:

```
show mgmt-acl-ipv6 name <NAME>
```

### *Syntax Description*

<code>show</code>	Display information
<code>mgmt-acl</code>	Display management ACL settings
<code>mgmt-acl-ipv6</code>	Display IPv6 management ACL settings
<code>all</code>	All configured management ACLs
<code>name</code>	Display information about a specific management ACL

---

<NAME>	The name of the management ACL
--------	--------------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus# show mgmt-acl all
Management ACL:
Name= New Name
  Restriction Type= range
  IP range= 192.168.11.1-192.168.11.253

Name= Remote 1
  Restriction Type= single
  IP address= 172.17.17.150

Name= Remote admin 2
  Restriction Type= single
  IP address= 172.17.16.12

ruckus#
```

## **Show Static Route Commands**

Use the `static-route` commands to display information about static routes configured on the controller.

### **show static-route all**

To display all static route information, use the following command:

```
show static-route all
```

### **show static-route name**

```
show static-route name <NAME>
```

### **show static-route-ipv6 all**

```
show static-route-ipv6 all
```

## show static-route-ipv6 name

```
show static-route-ipv6 name <NAME>
```

### Syntax Description

show	Display information
static-route	Display static route settings
static-route-ipv6	Display IPv6 static route settings
all	All configured static routes
name	Display information about a specific configured static route
<NAME>	The name of the static route entry

### Defaults

None.

### Example

```
ruckus# show static-route all
Static Route:
ID= 1
Name= Static Route 1
IP subnet= 192.168.11.1/24
IP gateway= 192.168.11.1
```

```
ruckus#
```

## Show WLAN Commands

Use the following commands to display information about available WLANs on the controller.

### show wlan

To display all available WLAN services (SSIDs), use the following command:

```
show wlan [all|name] <WORD>
```



## Syntax Description

show	Display information
wlan	Display WLAN services (SSIDs) settings
all	Display all WLAN services
name <WORD>	Display the named WLAN only

## Defaults

None.

## Example

```
ruckus(config)# show wlan all
WLAN Service:
  ID:
    1:
      NAME = Ruckus-WPA2
      Tx. Rate of Management Frame (2.4GHz) = 2.0Mbps
      Tx. Rate of Management Frame (5GHz) = 6.0Mbps
      Beacon Interval = 100ms
      SSID = Ruckus-WPA2
      Description = Ruckus-WPA2
      Type = Standard Usage
      Authentication = open
      Encryption = wpa2
      Algorithm = aes
      Passphrase = 10Asha10
      FT Roaming = Disabled
      802.11k Neighbor report = Disabled
      Web Authentication = Disabled
      Authentication Server = Disabled
      Called-Station-Id type = wlan-bssid
      Tunnel Mode = Disabled
      Background Scanning = Enabled
      Max. Clients = 100
      Isolation per AP = Disabled
      Isolation across AP = Disabled
      Zero-IT Activation = Enabled
      Priority = High
```

```
Load Balancing = Enabled
Band Balancing = Enabled
Dynamic PSK = Enabled
Dynamic PSK Passphrase Length = 62
Dynamic PSK Type = friendly
Dynamic PSK Expire Time = one-day
Dynamic PSK Validity Period = first-use
Limit Dynamic PSK = Disabled
Rate Limiting Uplink = Disabled
Rate Limiting Downlink = Disabled
Auto-Proxy configuration:
    Status = Disabled
Inactivity Timeout:
    Status = Enabled
    Timeout = 5 Minutes
VLAN-ID = 1
Dynamic VLAN = Disabled
Closed System = Disabled
Https Redirection = Disabled
OFDM-Only State = Disabled
Multicast Filter State = Disabled
802.11d State = Enabled
Force DHCP State = Disabled
Force DHCP Timeout = 10
DHCP Option82:
    Status = Disabled
    Option82 sub-Option1 = Disabled
    Option82 sub-Option2 = Disabled
    Option82 sub-Option150 = Disabled
    Option82 sub-Option151 = Disabled
Ignore unauthorized client statistic = Disabled
STA Info Extraction State = Enabled
BSS Minrate = Disabled
Call Admission Control State = Disabled
PMK Cache Timeout= 720 minutes
PMK Cache for Reconnect= Enabled
NAS-ID Type= wlan-bssid
Roaming Acct-Interim-Update= Disabled
PAP Message Authenticator = Enabled
```

```

Send EAP-Failure = Disabled
L2/MAC = No ACLS
L3/L4/IP Address = No ACLS
L3/L4/IPv6 Address = No ACLS
Precedence = Default
Proxy ARP = Disabled
Device Policy = No ACLS
Vlan Pool = No Pools
Role based Access Control Policy = Disabled
SmartRoam = Disabled Roam-factor = 1
White List = No ACLS
Application Visibility = enabled
Apply Policy Group = No_Denys

```

```
ruckus(config)#
```

## Show WLAN Group Commands

Use the following commands to display information about the WLAN groups that exist on the controller.

### **show wlan-group all**

To display a list of existing WLAN groups, use the following command:

```
show wlan-group all
```

### **Syntax Description**

show	Display information
wlan-group	Display information about the specified WLAN group
all	Show all WLAN groups

### **Defaults**

None.

### **Example**

```
ruckus# show wlan-group all
```

```

WLAN Group:
ID:
1:
Name= Default
Description= Default WLANs for Access Points
WLAN Service:
WLAN1:
NAME= Ruckus1
VLAN=
WLAN2:
NAME= Ruckus2
VLAN=

2:
Name= Guest WLAN Group
Description= 1st floor APs only
WLAN Service:
WLAN1:
NAME= Ruckus-Guest
VLAN=

ruckus#

```

## show wlan-group name

To display information about the specified WLAN group name, use the following command:

```
show wlan-group name <WORD>
```

### ***Syntax Description***

show	Display information
wlan-group name	Display information about the specified WLAN group name
<WORD>	The name of the WLAN group

## Defaults

None.

## Example

```
ruckus# show wlan-group name Default
WLAN Group:
ID:
1:
Name= Default
Description= Default WLANs for Access Points
WLAN Service:
WLAN1:
NAME= Ruckus1
VLAN=
WLAN2:
NAME= Ruckus2
VLAN=

ruckus#
```

## Show L2 Access Control List Commands

Use the `show l2acl` commands to display Layer 2 access control list rules that have been added to the controller.

### show l2acl all

To display all Layer 2 access control list (ACL) rules that have been added to the controller and their settings, use the following command:

```
show l2acl all
```

### Syntax Description

<code>show</code>	Display information
<code>l2acl</code>	Display L2 ACL information
<code>all</code>	All L2 ACL

## **Defaults**

None.

## **Example**

```
ruckus# show l2acl all
```

L2/MAC ACL:

ID:

1:

Name= System

Description= System

Restriction: Deny only the stations listed below

Stations:

2:

Name= blocked-sta-list

Description=

Restriction: Deny only the stations listed below

Stations:

## **show l2acl name**

To display the settings of a specific L2 ACL rule that has been added to the controller, use the following command:

```
show l2acl name <WORD>
```

## **Syntax Description**

show	Display information
l2acl	Display L2 ACL information
name	Display information about the specified L2 ACL rule name
<WORD>	Name of the L2 ACL rule

## **Defaults**

None.

**Example**

```
ruckus# show l2acl name 1
L2/MAC ACL:
ID:
2:
Name= 1
Description=
Restriction: Deny only the stations listed below
Stations:
MAC Address= 00:33:22:45:34:88
```

**Show Whitelist Commands**

Use the `show whitelist` commands to display client isolation whitelists that have been added to the controller.

**show whitelist all**

To display all whitelists that have been added to the controller and their settings, use the following command:

```
show whitelist all
```

**Syntax Description**

<code>show</code>	Display information
<code>whitelist</code>	Display whitelist information
<code>all</code>	All whitelists

**Defaults**

None.

**Example**

```
ruckus# show whitelist all
White Lists:
ID:
```

```

1:
  Name= printer whitelist
  Description= printer
  Rules:
    1:
      Description= printer
      MAC = 12:34:56:78:90:00
      IP Address = 192.168.4.10

ruckus#

```

## show whitelist name

To display a specified whitelist that has been added to the controller by name, use the following command:

```
show whitelist name <WORD>
```

### Syntax Description

show	Display information
whitelist	Display whitelist information
name <WORD>	Specify the name of the whitelist

### Defaults

None.

### Example

```

ruckus# show whitelist name "printer whitelist"
White Lists:
  ID:
    1:
      Name= printer whitelist
      Description= printer
      Rules:
        1:
          Description= printer

```



```
MAC = 12:34:56:78:90:00  
IP Address = 192.168.4.10
```

```
ruckus#
```

## Show L3 Access Control List Commands

Use the `show l3acl` commands to display Layer 3 access control list rules that have been added to the controller.

### show l3acl all

To display all Layer 3 access control list (ACL) rules that have been added to the controller and their settings, use the following command:

```
show l3acl all
```

### show l3acl-ipv6 all

To display all IPv6 Layer 3 access control list (ACL) rules that have been added to the controller and their settings, use the following command:

```
show l3acl-ipv6 all
```

### Syntax Description

<code>show</code>	Display information
<code>l3acl</code>	Display L3 ACL information
<code>l3acl-ipv6</code>	Display IPv6 L3 ACL information
<code>all</code>	All L3 ACL

### Defaults

None.

### Example

```
ruckus# show l3acl all  
L3/L4/IP ACL:  
ID:  
4:  
Name= test2
```

```
Description= test2
Default Action if no rule is matched= Deny all by default
Rules:
Order= 1
Description=
Type= Allow
Destination Address= Any
Destination Port= 53
Protocol= Any
Order= 2
Description=
Type= Allow
Destination Address= Any
Destination Port= 67
Protocol= Any
Order= 3
Description=
Type= Allow
Destination Address= 8.8.8.8/24
Destination Port= 25
Protocol= 6
```

### **show l3acl name**

To display the settings of a specific L3 ACL rule that has been added to the controller, use the following command:

```
show l3acl name <WORD>
```

### **show l3acl-ipv6 name**

To display the settings of a specific IPv6 L3 ACL rule that has been added to the controller, use the following command:

```
show l3acl-ipv6 name <WORD>
```

### ***Syntax Description***

---

show	Display information
l3acl	Display L3 ACL information
l3acl-ipv6	Display IPv6 L3 ACL information

---

---

name	Display information about the specified L3 ACL rule
<WORD>	Name of the L3 ACL rule

---

### **Defaults**

None.

### **Example**

```
ruckus# show l3acl name test2
L3/L4/IP ACL:
ID:
4:
Name= test2
Description= test2
Default Action if no rule is matched= Allow all by default
Rules:
Order= 1
Description=
Type= Allow
Destination Address= Any
Destination Port= 53
Protocol= Any
Order= 2
Description=
Type= Allow
Destination Address= Any
Destination Port= 67
Protocol= Any
Order= 3
Description=
Type= Allow
Destination Address= 8.8.8.8/24
Destination Port= 25
Protocol= 6
```

## **Show Hotspot Commands**

Use the `show hotspot` commands to display the controller's hotspot configuration settings.

## show hotspot all

To display a list of all hotspot services that have been created on the controller, use the following command:

```
show hotspot all
```

### Syntax Description

show	Display information
hotspot	Display hotspot information
all	All available hotspots

### Defaults

None.

### Example

```
ruckus# show hotspot all
Hotspot:
  ID:
    1:
      Name= Hotspot 1
      WISPr Smart Client Support:
        Status= None
      Login Page Url= http://192.168.1.12/login.htm
      Start Page= redirect to the URL that the user intends
to visit
      Session Timeout:
        Status= Disabled
      Grace Period:
        Status= Disabled
      Intrusion Prevention= Enabled
      Authentication Server= Local Database
      Accounting Server:
        Status= Disabled
      Isolation per AP = Disabled
      Isolation across AP = Disabled
      White List = No ACLS
      Location ID=
      Location Name=
```

```
Walled Garden 1= 1.1.1.1
```

```
IPv4 Rules:
```

```
IPv6 Rules:
```

```
ruckus#
```

## show hotspot name

To display information about the specific hotspot service, use the following command:

```
show hotspot name <WORD>
```

If the hotspot name includes a space, you must put the name in quotation marks (for example, "hotspot name").

### Syntax Description

show	Display information
hotspot name	Display hotspot information
<WORD>	The name of the hotspot

### Defaults

None.

### Example

```
ruckus# show hotspot name "Hotspot 1"
```

```
Hotspot:
```

```
ID:
```

```
1:
```

```
Name= Hotspot 1
```

```
WISPr Smart Client Support:
```

```
Status= None
```

```
Login Page Url= http://192.168.1.12/login.htm
```

```
Start Page= redirect to the URL that the user intends to visit
```

```
Session Timeout:
```

```
Status= Disabled
```

```

Grace Period:
  Status= Disabled
Intrusion Prevention= Enabled
Authentication Server= Local Database
Accounting Server:
  Status= Disabled
Isolation per AP = Disabled
Isolation across AP = Disabled
White List = No ACLS
Location ID=
Location Name=
Walled Garden 1= 1.1.1.1
IPv4 Rules:

IPv6 Rules:

```

```
ruckus#
```

## show hs20op all

To display information about all Hotspot 2.0 Operators, use the following command:

```
show hs20op all
```

### Syntax Description

show	Display information
hs20op	Display Hotspot 2.0 Operator
all	Display all HS2.0 operators

### Defaults

None.

### Example

```

ruckus# show hs20op all
Hotspot 2.0 Operator:
  ID:
  1:

```

```
NAME= operator1
Description=
Venue Group= Unspecified
Venue Type= Unspecified
ASRA Option:
    Status= Disabled
Internet Option= Disabled
Access Network Type= Private
IPv4 Address Type= Not Available
IPv6 Address Type= Not Available
HESSID=
Friendly Name List:
Service Provider Profiles:
    ID= 1
        Name= provider1
WAN Metrics:
    Enable Symmetric Link= Disabled
    WAN at Capability= Disabled
    Link Status= Link Up
    WAN Downlink Load= 0
    WAN Downlink Speed= 0
    WAN Uplink Load= 0
    WAN Uplink Speed= 0
    Load Measurement Duration= 0
Connection Capability:
    Description= ICMP
        IP Protocol= 1
        Port Number= 0
        Status= Closed
    Description= FTP
        IP Protocol= 6
        Port Number= 20
        Status= Closed
    Description= SSH
        IP Protocol= 6
        Port Number= 22
        Status= Closed
    Description= HTTP
        IP Protocol= 6
```

```
Port Number= 80
  Status= Closed
Description= Used by TLS VPNs
  IP Protocol= 6
  Port Number= 443
  Status= Closed
Description= Used by PPTP VPNs
  IP Protocol= 6
  Port Number= 1723
  Status= Closed
Description= VoIP
  IP Protocol= 6
  Port Number= 5060
  Status= Closed
Description= Used by IKEv2 (IPSec VPN)
  IP Protocol= 17
  Port Number= 500
  Status= Closed
Description= VoIP
  IP Protocol= 17
  Port Number= 5060
  Status= Closed
Description= May be used by IKEv2 (IPSec VPN)
  IP Protocol= 17
  Port Number= 4500
  Status= Closed
Description= ESP, used by IPSec VPNs
  IP Protocol= 50
  Port Number= 0
  Status= Closed
Additional Connection Capability:
Advanced GAS Settings:
  GAS query response buffering time= 1000
  GAS DOS detection= Disabled
  GAS DOS maximum request number= 200
Hotspot 2.0 Capability:
  Operating Class Indication= Unspecified
```



```
ruckus#
```

## show hs20op name

To display information about the named Hotspot 2.0 Operator, use the following command:

```
show hs20op name <WORD>
```

### Syntax Description

show	Display information
hs20op name	Display specific Hotspot 2.0 Operator
<WORD>	The name of the HS2.0 operator

### Defaults

None.

### Example

```
ruckus# show hs20op name operator1
```

```
Hotspot 2.0 Operator:
```

```
ID:
```

```
1:
```

```
NAME= operator1
```

```
Description=
```

```
Venue Group= Unspecified
```

```
Venue Type= Unspecified
```

```
ASRA Option:
```

```
Status= Disabled
```

```
Internet Option= Disabled
```

```
Access Network Type= Private
```

```
IPv4 Address Type= Not Available
```

```
IPv6 Address Type= Not Available
```

```
HESSID=
```

```
Friendly Name List:
```

```
Service Provider Profiles:
```

```
ID= 1
```

```
Name= provider1
```

```
WAN Metrics:
```

```
Enable Symmetric Link= Disabled
```

```
WAN at Capability= Disabled
Link Status= Link Up
WAN Downlink Load= 0
WAN Downlink Speed= 0
WAN Uplink Load= 0
WAN Uplink Speed= 0
Load Measurement Duration= 0
Connection Capability:
Description= ICMP
  IP Protocol= 1
  Port Number= 0
  Status= Closed
Description= FTP
  IP Protocol= 6
  Port Number= 20
  Status= Closed
Description= SSH
  IP Protocol= 6
  Port Number= 22
  Status= Closed
Description= HTTP
  IP Protocol= 6
  Port Number= 80
  Status= Closed
Description= Used by TLS VPNs
  IP Protocol= 6
  Port Number= 443
  Status= Closed
Description= Used by PPTP VPNs
  IP Protocol= 6
  Port Number= 1723
  Status= Closed
Description= VoIP
  IP Protocol= 6
  Port Number= 5060
  Status= Closed
Description= Used by IKEv2 (IPSec VPN)
  IP Protocol= 17
  Port Number= 500
```

```

        Status= Closed
Description= VoIP
        IP Protocol= 17
        Port Number= 5060
        Status= Closed
Description= May be used by IKEv2 (IPSec VPN)
        IP Protocol= 17
        Port Number= 4500
        Status= Closed
Description= ESP, used by IPSec VPNs
        IP Protocol= 50
        Port Number= 0
        Status= Closed
Additional Connection Capability:
Advanced GAS Settings:
        GAS query response buffering time= 1000
        GAS DOS detection= Disabled
        GAS DOS maximum request number= 200
Hotspot 2.0 Capability:
        Operating Class Indication= Unspecified

```

ruckus#

## show hs20sp all

To display information about the Hotspot 2.0 Service Provider, use the following command:

```
show hs20sp all
```

### ***Syntax Description***

show	Display information
hs20sp	Display Hotspot 2.0 Service Provider
all	Display all HS2.0 Service Providers

### ***Defaults***

None.

**Example**

```
ruckus# show hs20sp all
Hotspot 2.0 Service Provider:
  ID:
    1:
      NAME= provider1
      Description=
      Realm List:
      Domain Name List:
      Roaming Consortium List:
      3GPP Cellular Network information:
```

```
ruckus#
```

**show hs20sp name**

To display information about a specific Hotspot 2.0 Service Provider, use the following command:

```
show hs20sp name <WORD>
```

**Syntax Description**

show	Display information
hs20sp name	Display specific Hotspot 2.0 Service Provider
<WORD>	The name of the HS2.0 Service Provider

**Defaults**

None.

**Example**

```
ruckus# show hs20sp name provider1
Hotspot 2.0 Service Provider:
  ID:
    1:
      NAME= provider1
      Description=
      Realm List:
```

```
Domain Name List:  
Roaming Consortium List:  
3GPP Cellular Network information:
```

```
ruckus#
```

## Show Guest Policy Commands

Use the following commands to display guest access services.

### show guest-access-service

To display a list of guest access services or a specific service, use the following command:

```
show guest-access-service [all|name <WORD>]
```

### Example

```
ruckus# show guest-access all
```

```
Guest Access:
```

```
Name = guestpolicy1
```

```
Onboarding Portal:
```

```
Aspect = Guest pass and ZeroIT
```

```
Authentication:
```

```
Mode = Use guest pass authentication
```

```
Multiple users to share a single guest pass = Disallowed
```

```
Title = hello
```

```
Terms of Use:
```

```
Status = Disabled
```

```
Redirection:
```

```
Mode = To the URL that the user intends to visit
```

```
Restricted Subnet Access:
```

```
Rules:
```

```
1:
```

```
Description=
```

```
Type= Deny
```

```
Destination Address= local
```

```
Destination Port= Any
```

```
Protocol= Any
```

```
2:
  Description=
  Type= Deny
  Destination Address= 10.0.0.0/8
  Destination Port= Any
  Protocol= Any
3:
  Description=
  Type= Deny
  Destination Address= 172.16.0.0/12
  Destination Port= Any
  Protocol= Any
4:
  Description=
  Type= Deny
  Destination Address= 192.168.0.0/16
  Destination Port= Any
  Protocol= Any
```

Restricted IPv6 Access:

Rules:

```
1:
  Description=
  Type= Deny
  Destination Address= local
  Destination Port= Any
  Protocol= Any
  ICMPv6 Type= Any
```

ruckus#

## Show Hotspot 2.0 Operator Commands

Use the following commands to display Hotspot 2.0 Operators.

### **show hs20op**

To display a list of Hotspot 2.0 operators, use the following command:

```
show hs20op [all|name <WORD>]
```

### **Example**

```
ruckus# show hs20op all
```

## **Show Hotspot 2.0 Service Provider Commands**

Use the following commands to display Hotspot 2.0 Service Providers.

### **show hs20sp**

To display a list of Hotspot 2.0 service providers, use the following command:

```
show hs20sp [all|name <WORD>]
```

### **Example**

```
ruckus# show hs20sp all
```

## **Show Role Commands**

Use the `show role` commands to display details about roles that have been created on the controller.

### **show role all**

To display a list of all roles that have been created on the controller, use the following command:

```
show role all
```

### **Syntax Description**

---

<code>show</code>	Display information
<code>role</code>	Display role information
<code>all</code>	All roles that have been created

---

### **Defaults**

None.

**Example**

```
ruckus# show role all
Role:
  ID:
    1:
      Name= Default
      Description= Allow Access to All WLANs
      Group Attributes=
      Guest Pass Generation= Allowed
      ZoneDirector Administration:
        Status= Allowed
        Allow ZoneDirector Administration= Super Admin
      Allow All WLANs:
        Mode= Allow access to all WLANs
        Access Control Policy= Disallowed

ruckus#
```

**show role name**

To display information about the specific role, use the following command:

```
show role name <WORD>
```

**Syntax Description**

show	Display information
role name	Display role information
<WORD>	The name of the role

**Defaults**

None.

**Example**

```
ruckus# show role name Default
Role:
  ID:
```



```
1:
  Name= Default
  Description= Allow Access to All WLANs
  Group Attributes=
  Guest Pass Generation= Allowed
  ZoneDirector Administration:
    Status= Allowed
    Allow ZoneDirector Administration= Super Admin
  Allow All WLANs:
    Mode= Allow access to all WLANs
  Access Control Policy= Disallowed

ruckus#
```

## Show VLAN Pool Commands

Use the following commands to display VLAN pools.

### show vlan-pool

To display a list of VLAN pools, use the following command:

```
show vlan-pool [all|name <WORD>]
```

### **Example**

```
ruckus# show vlan-pool all
VLAN Pool:
  ID:
    1:
      Name = vlan pool 1
      Description =
      Option = 1
      VLANSET = 10,20,30,40,50-55

ruckus#
```

# Show User Commands

Use the `show user` commands to display details about user accounts that exist on the controller.

## show user all

To display a list of all existing user accounts, use the following command:

```
show user all
```

### Syntax Description

<code>show</code>	Display information
<code>user</code>	Display user information
<code>all</code>	All existing user accounts

### Defaults

None.

### Example

```
ruckus# show user all
User:
ID:
1:
User Name= test22
Full Name= test11
Password= test1234
Role= Default
```

## show user name

To display information about the specific user, use the following command:

```
show user name <user_name>
```

### Syntax Description

<code>show</code>	Display information
<code>user name</code>	Display user information
<code>&lt;WORD&gt;</code>	The name of the user

## Defaults

None.

## Example

```
ruckus# show user name test22
User:
ID:
1:
User Name= test22
Full Name= test11
Password= test1234
Role= Default
```

# Show Currently Active Clients Commands

Use the `show current-active-clients` commands to display a list of wireless clients that are associated with the APs that the controller manages.

## show current-active-clients all

To display a list of all existing user accounts, use the following command:

```
show current-active-clients all
```

## Syntax Description

show	Display information
current-active-clients	Display currently active wireless clients
all	All active wireless clients

## Defaults

None.

## Example

```
ruckus# show current-active-clients all
Current Active Clients:
Clients:
Mac Address= 00:22:fb:5c:e2:32
```

```
User/IP= 172.18.30.2
User/IPv6=
Access Point= 04:4f:aa:13:30:f0
BSSID= 04:4f:aa:13:30:fa
Connect Since=2011/03/01 02:48:22
Auth Method= OPEN
WLAN= 11jojoe
VLAN= None
Channel= 6
Radio= 802.
Signal= 0
Status= Authorized
```

Last 300 Events/Activities:

Activity:

Date/Time= 2011/03/01 02:49:05

Severity= Low

User=

Activities= User[00:22:fb:5c:e2:32] joins WLAN[11jojoe] from  
AP[04:4f:aa:13:30:f0]

Activity:

Date/Time= 2011/03/01 02:48:22

Severity= Low

User=

Activities= User[00:22:fb:5c:e2:32] joins WLAN[11jojoe] from  
AP[04:4f:aa:13:30:f0]

...

...

ruckus#

## **show current-active-clients mac**

To display information about the specific active client, use the following command:

```
show current-active-clients mac <MAC>
```

### ***Syntax Description***

---

show	Display information
------	---------------------

---

---

current-active- Display currently active wireless clients  
clients mac

---

<MAC> The MAC address of the wireless client

---

### **Defaults**

None.

### **Example**

```
ruckus# show current-active-clients mac 6c:62:6d:1b:e3:00
Current Active Clients:
Clients:
Mac Address= 6c:62:6d:1b:e3:00
User/IP= 192.168.11.11
User/IPv6=
Access Point= 04:4f:aa:0c:b1:00
BSSID= 04:4f:aa:0c:b1:08
Connect Since=2012/01/10 06:22:44
Auth Method= OPEN
WLAN= Ruckus1
VLAN= None
Channel= 6
Radio= 802.11gn
Signal= 53
Status= Authorized
Received from client= 20746 pkts / 6274531 bytes
Transmitted to client= 25777 pkts / 6714433 bytes
Tx. drops due to retry failure= 1 pkts

Last 300 Events/Activities:
Activity:
Date/Time= 2012/01/10 06:22:44
Severity= Low
User=
Activities= User[6c:62:6d:1b:e3:00] joins WLAN[Ruckus1] from
AP[7962 - MAP@04:4f:aa:0c:b1:00]
Activity:
Date/Time= 2012/01/09 18:52:28
Severity= Low
```

```

User=
Activities= User[6c:62:6d:1b:e3:00] disconnects from WLAN[Ruckus1]
at AP[7363 - RAP@00:24:82:3f:14:60]
Activitiy:
Date/Time= 2012/01/08 06:08:52
Severity= Low
User=
Activities= AP[7363 - RAP@00:24:82:3f:14:60] radio [11g/n] detects
User[6c:62:6d:1b:e3:00] in WLAN[Ruckus1] roams from AP[7962 -
MAP@04:4f:aa:0c:b1:00]
...
...
ruckus#

```

## Show Mesh Commands

Use the `show mesh` commands to display the controller's mesh network configuration and topology.

### show mesh info

To display a list of all mesh networks that have been formed, use the following command:

```
show mesh info
```

### Syntax Description

<code>show</code>	Display information
<code>mesh</code>	Display mesh network information
<code>info</code>	Show mesh information

### Defaults

None.

### Example

```

ruckus# show mesh info
Mesh Settings:
Mesh Status= Enabled
Mesh Name (ESSID)= Mesh-000000000311

```

```

Mesh Passphrase= GdxW5CUgrn_SEHOPyCSxv_cQHScA MH-OpnRGfX sRvwXBJL-
wUsD6eeK8CMEZfm
Mesh Hop Detection:
Status= Disabled
Mesh Downlinks Detection:
Status= Disabled
Tx. Rate of Management Frame=2Mbps
Beacon Interval= 200ms
ruckus#

```

## show mesh topology

To display the topology of existing mesh networks, use the following command:

```
show mesh topology
```

### Syntax Description

show	Display information
mesh	Display mesh network information
topology	Show mesh topology

### Defaults

None.

### Example

```

ruckus# show mesh topology
Mesh Topology(Mesh-000000000311):
Root Access Points= 00:24:82:3b:14:60
Signal (dB) Downlink=/ Uplink=
Description= 7363 - RAP (Study)
Channel= 153 (11an)
IP Address= 192.168.11.3
Mesh Access Points= 04:4f:ab:0c:b1:00
Signal (dB) Downlink= 28 / Uplink= 30
Description= 7962 MAP (Living Room)
Channel= 153
IP Address= 192.168.11.6

```

```
ruckus#
```

## Show Dynamic PSK Commands

Use the `show dynamic-psks` commands to display information about Dynamic PSKs that have been generated. Use the following command:

```
show dynamic-psks
```

### Syntax Description

---

<code>show</code>	Display information
<code>dynamic-psks</code>	Display dynamic PSKs that have been generated

---

### Defaults

None.

### Example

```
ruckus# show dynamic-psks
Generated Dynamic PSKs:
DPSK:
User= BatchDPSK_User_1
Mac Address= 00:00:00:00:00:00
Created= 2011/03/01 03:30:01
Expired= Unlimited
DPSK:
User= BatchDPSK_User_2
Mac Address= 00:00:00:00:00:00
Created= 2011/03/01 03:30:02
Expired= Unlimited
DPSK:
User= DPSK-User-2
Mac Address= 00:11:22:33:44:55
Created= 2011/03/01 03:30:47
Expired= Unlimited
```



## Show Dynamic Certificate Commands

Use the `show dynamic-certs` commands to display information about Dynamic certificates that have been generated. Use the following command:

```
show dynamic-certs
```

### Syntax Description

---

<code>show</code>	Display information
<code>dynamic-certs</code>	Display dynamic certificates that have been generated

---

### Defaults

None.

### Example

```
ruckus# show dynamic-certs  
Generated Dynamic Certs:
```

## Show Guest Pass Commands

Use the `show guest-passes` commands to display information about guest passes that have been generated. Use the following command:

```
show guest-passes
```

### Syntax Description

---

<code>show</code>	Display information
<code>guest-passes</code>	Display guest passes that have been generated

---

### Defaults

None.

### Example

```
ruckus# show guest-passes  
Generated Guest Passes:  
ID:  
Guest Name= John Doe  
Remarks=
```

```
Expires= 2012/01/11 08:32:15
Re-auth=
Creator= ruckus
Sharable= No
Wlan= Ruckus-Guest

ruckus#
```

## Show Rogue Device Commands

Use the `show rogue-devices` commands to display information about rogue devices that the controller has detected on the network. Use the following command:

```
show rogue-devices
```

### Syntax Description

<code>show</code>	Display information
<code>rogue-devices</code>	Display rogues devices that have been detected on the network

### Defaults

None.

### Example

```
ruckus# show rogue-devices
Current Active Rogue Devices:
Rogue Devices:
Mac Address= 00:25:c4:52:1c:a1
Channel= 6
Radio= 802.11bg
Type= AP
Encryption= Open
SSID= V54-HOME001
Last Detected= 2011/03/01 02:03:43

Known/Recognized Rogue Devices:
```

## Show Events and Activities Commands

Use the `show events-activities` commands to display information events and network activities that have been recorded by the controller. Use the following command:

```
show events-activities
```

### Syntax Description

---

<code>show</code>	Display information
<code>events-activities</code>	Display a list of events and activities records by the controller

---

### Defaults

None.

### Example

```
ruckus# show events-activities
ruckus# show events-activities
Last 300 Events/Activities:
Activity:
Date/Time= 2012/01/10 08:33:17
Severity= Low
User=
Activities= Admin[ruckus] logs in from [192.168.11.7]
Activity:
Date/Time= 2012/01/10 08:32:00
Severity= Low
User=
Activities= WLAN[Ruckus-Guest] with BSSID[04:4f:aa:4c:b1:08]
configuration has been updated on radio [11g/n] of AP[7962 -
MAP@04:4f:aa:0c:b1:00]
Activity:
Date/Time= 2012/01/10 08:32:00
Severity= Low
User=
...
...
```

## Show Alarm Commands

Use the `show alarm` commands to display alarms that have been generated by the controller. Use the following command:

```
show alarm
```

### Syntax Description

---

<code>show</code>	Display information
<code>alarm</code>	Display a list of alarms that have been generated by the controller

---

### Defaults

None.

### Example

```
ruckus# show alarm
Last 300 Alarms:
  Alarms:
    Date/Time= 2013/03/27 15:36:59
    Name= AP Lost Contact
    Severity= High
    Activities= Lost contact with AP[7372 - MAP@c0:c5:20:3b:91:f0]
  Alarms:
    Date/Time= 2013/03/18 14:44:21
    Name= ZD warm restart
    Severity= Medium
    Activities= System warm restarted with [user reboot].
...
...
ruckus#
```

## Show License Commands

Use the `show license` commands to display the controller's license information, including the model number, the maximum number of APs that it can support, and the maximum number of wireless clients that managed APs can support. Use the following command:

```
show license
```

### ***Syntax Description***

show	Display information
license	Display the controller's license information

### ***Defaults***

None.

### ***Example***

```
ruckus# show license
License:
  Model= ZD1112
  Max. AP Number= 12
  Max. Client Number= 1250
ruckus#
```

## **Show USB Software Commands**

Use the show usb-software command to display current USB software package information.

### **show usb-software**

```
show usb-software
```

### ***Syntax Description***

show	Display information
usb-software	Display USB software package information

### ***Defaults***

None.

### ***Example***

```
ruckus# show usb-software
Sorry, the USB Software hasn't been found.
```

```
ruckus#
```

## Show Application Denial Policy Commands

Use the following commands to display application denial policies, user-defined applications and application port-mapping settings.

### show app-denial-policy

Displays the application denial policy settings.

#### Example

```
ruckus# show app-denial-policy
Application Denial Policy:
  ID:
    1:
      Name= facebook
      Description= deny facebook
      Default Mode= accept
      Rules:
        1:
          Application= HTTP hostname
          Description= facebook.com
ruckus#
```

### show user-defined-app

Displays the user defined application settings.

#### Example

```
ruckus# show user-defined-app
User Defined Application:
  ID:
    1:
      Application= angry birds
      DST-IP= 216.146.46.10
      Netmask= 255.255.255.0
      DST-Port= 5050
```

```
Protocol= tcp
ruckus#
```

## show app-port-mapping

Displays the application category mapping settings.

### Example

```
ruckus# show app-port-mapping
Application Port Mapping:
  ID:
  1:
    Name= 2100-tcp
    Port= 2100
    Protocol= tcp
    Description= Facebook
ruckus#
```

## Show Session-Timeout Commands

Use the `show session-timeout` command to display the current session timeout interval.

### show session-timeout

```
show session-timeout
```

### Syntax Description

---

<code>show</code>	Display information
<code>session-timeout</code>	Display the current session timeout interval

---

### Defaults

None.

### Example

```
ruckus# show session-timeout
Current session timeout interval is 30 minutes
```

```
ruckus#
```

## Show Active Wired Client Commands

Use the `show active-wired-client` commands to display information about currently active wired clients.

### show active-wired-client all

```
show active-wired-client all
```

### show active-wired-client mac

```
show active-wired-client mac <MAC>
```

### Syntax Description

<code>show</code>	Display information
<code>active-wired-client</code>	Display the currently active wired client information
<code>all</code>	Show all wired clients
<code>mac</code>	Show a specific client information by MAC address
<code>&lt;MAC&gt;</code>	The MAC address of the specific client

### Defaults

None.

### Example

```
ruckus# show active-wired-client all
```

```
Current Active Wired Clients:
```

```
ruckus#
```

## Show RADIUS Statistics Commands

Use the following commands to display RADIUS statistics or to reset RADIUS statistics.



## show radius-statistics

To display a list of RADIUS server statistics, use the following command:

```
show radius-statistics [server-all|server-
name<WORD>] | [wlan-all|wlan-name<NAME>] [latest-ten-
min|latest-one-hour|latest-one-day]
```

### Syntax Description

show radius-statistics	Display list of RADIUS server statistics.
server-all	Display statistics for all servers. (Default: recorded from power on.)
server-name <WORD>	Display statistics for the specified server. (Default: recorded from power on.)
wlan-all	Display statistics for all WLANs. (Default: recorded for the last day.)
wlan-name <NAME>	Display statistics for the specified WLAN. (Default: recorded for the last day.)
latest-ten-min	Display statistics for the last 10 minutes.
latest-one-hour	Display statistics for the last hour.
latest-one-day	Display statistics for the last day.

## reset radius-statistics

To reset RADIUS statistics, use the following command:

```
reset radius-statistics [server-all|server-
name<WORD>] [master|standby] [latest-ten-min|latest-one-
hour|latest-one-day]
```

### Syntax Description

reset radius-statistics	Reset RADIUS server statistics.
server-all	Reset statistics for all servers to zero. (Default: recorded from power on.)
server-name <WORD>	Reset statistics for the specified server to zero. (Default: recorded from power on.)

wlan-all	Reset statistics for all WLANs. (Default: recorded for the last day.)
wlan-name <NAME>	Reset statistics for the specified WLAN. (Default: recorded for the last day.)
master	Reset statistics of the master server to zero.
standby	Reset statistics of the standby server to zero.
latest-ten-min	Reset statistics recorded for the last 10 minutes
latest-one-hour	Reset statistics recorded for the last hour
latest-one-day	Reset statistics recorded for the last day

## Show Load Balancing Commands

Use the following commands to display AP load balancing information.

### show load-balance

To display AP load balancing information, use the following command:

```
show load-balance
```

### Example

```
ruckus# show load-balance
*** Show AP load balance
Radio---Enable--Scan--ActThresh---AdjThresh---WeakBypass---
StrongBypass---NewActTrigger---Headroom
2GHz      0  2000      10      50      33      55
3
5GHz      0  2000      10      43      35      55
3
----MAC Address----Cli-New-Lim---Allow-----Fallbk----Adjacent 2-
GHz Radios [MacAdrs FwdRssi RevRssi SumRssi]
c4:10:8a:1f:d1:f0  1  0  0 1000000000 0000000000
c0:c5:20:3b:91:f0  2  0  0 1000000000 0000000000
----MAC Address----Cli-New-Lim---Allow-----Fallbk----Adjacent 5-
GHz Radios [MacAdrs FwdRssi RevRssi SumRssi]
c4:10:8a:1f:d1:f0  0  0  0 1000000000 0000000000
c0:c5:20:3b:91:f0  1  0  0 1000000000 0000000000
```

```
ruckus#
```

## Monitor AP MAC Commands

Use the `monitor ap mac` command to monitor details on a specific access point.

### monitor ap mac

```
monitor ap mac <MAC>
```

### Syntax Description

<code>monitor</code>	Begin monitoring mode
<code>ap mac</code>	Designate the access point to begin monitoring
<code>&lt;MAC&gt;</code>	The MAC address of the specific access point

### Defaults

None.

### Example

```
ruckus# monitor ap mac 04:4f:aa:0c:b1:00
```

```
-----
ID MAC Approved Device-Name Description
104:4f:aa:0c:b1:00 Yes7962 - MAP7962 MAP (Living)
-----
```

```
-----
IPv4-ADDRMASK GATEWAYPRI-DNS
192.168.11.6 255.255.255.0192.168.11.1
-----
```

```
-----
Radio-TypeRX-Packets (M) /RX-Bytes (G) TX-Packets (M) /TX-Bytes (G)
Retries(%)
```

```
Radio a/n 36.9/2.028.6/2.00.0
```

```
Radio-TypeRX-Packets (M) /RX-Bytes (G) TX-Packets (M) /TX-Bytes (G)
Retries(%)
```

```
Radio b/g/n 37.8/2.012.4/2.00.3
-----
```

```
Status Mode LocationUplink-Status
```

```
EnabledAuto Living Room Smart
```

```
-----
```

```
-----
```

```
-----
```

```
ID MAC Approved Device-Name Description
```

```
104:4f:aa:0c:b1:00 Yes7962 - MAP7962 MAP (Living
```

```
-----
```

```
-----
```

```
IPv4-ADDRMASK GATEWAYPRI-DNS
```

```
192.168.11.6 255.255.255.0192.168.11.1
```

```
-----
```

```
-----
```

```
Radio-TypeRX-Packets (M) /RX-Bytes (G) TX-Packets (M) /TX-Bytes (G)
```

```
Retries(%)
```

```
Radio a/n 36.9/2.028.6/2.00.0
```

```
Radio-TypeRX-Packets (M) /RX-Bytes (G) TX-Packets (M) /TX-Bytes (G)
```

```
Retries(%)
```

```
Radio b/g/n 37.8/2.012.4/2.00.3
```

```
-----
```

```
-----
```

```
Status Mode LocationUplink-Status
```

```
EnabledAuto Living Room Smart
```

```
-----
```

```
-----
```

```
-----
```

```
ID MAC Approved Device-Name Description
```

```
104:4f:aa:0c:b1:00 Yes7962 - MAP7962 MAP (Living
```

```
-----
```

```
-----
```

```
IPv4-ADDRMASK GATEWAYPRI-DNS
```

```
192.168.11.6 255.255.255.0192.168.11.1
```

```
-----
```

```
-----
```

```
Radio-TypeRX-Packets (M) /RX-Bytes (G) TX-Packets (M) /TX-Bytes (G)
```

```
Retries(%)
```

```
Radio a/n 36.9/2.028.6/2.00.0
```

```
Radio-TypeRX-Packets (M) /RX-Bytes (G) TX-Packets (M) /TX-Bytes (G)
```

```
Retries(%)
```

```
Radio b/g/n 37.8/2.012.4/2.00.3
```

```
-----  
-----  
Status Mode LocationUplink-Status  
EnabledAuto Living Room Smart  
-----  
-----
```

```
ruckus#
```

## Monitor Currently Active Client Commands

Use the `monitor current-active-clients` command to monitor details on a specific client.

### monitor current-active-clients

```
monitor current-active-clients mac <MAC>
```

### *Syntax Description*

<code>monitor</code>	Begin monitoring mode
<code>current-active-clients mac</code>	Designate the currently active client to begin monitoring
<code>&lt;MAC&gt;</code>	The MAC address of the specific client

### *Defaults*

None.

### *Example*

```
ruckus# monitor current-active-clients mac 00:22:fb:ad:1b:2e
```

```
-----  
-----  
04:4f:aa:0c:b1:00 192.168.11.7 Ruckus1 None Authorized  
-----  
-----
```

```
04:4f:aa:0c:b1:0c153 11an43 OPEN
```

```
-----
-----
44.3/6.743.2/17.0 36
-----
-----
-----
```

```
ruckus#
```

### **monitor current-active-clients-mcs-info**

To monitor MCS information for the specified current active clients, use the following command:

```
monitor current-active-clients-mcs-info sta-mac <MAC> ap-
mac <MAC> bssid <BSSID>
```

### ***Syntax Description***

monitor	Begin monitoring mode
current-active-clients-mcs-info	Monitor MCS info of currently active clients
sta-mac <MAC>	The MAC address of the specific client
ap-mac <MAC>	MAC address of the AP
bssid <BSSID>	Monitor clients connected to the specified BSSID

## **Monitor Sysinfo Commands**

Use the `monitor sysinfo` command to monitor system information.

### **monitor sysinfo**

```
monitor sysinfo
```

## Syntax Description

---

monitor	Begin monitoring mode
sysinfo	Display the system information

---

### Example

```
ruckus# monitor sysinfo
```

```
-----
-----
IPv4-ADDR IPv6-ADDR MAC Uptime Model MAX-APs
192.168.11.100NULL 00:13:11:01:01:01 12d 1h 29mZD111212
-----
-----
Number-of-APs Number-of-ClientsNumber-of-Rogues Name
2 10ruckus
-----
-----
Usage of 1 hr|Usage of 24 hr
Max-Concurrent-Users TX-BytesRogues | Max-Concurrent-Users TX-
BytesRogues
12.33M 02297.58M 2
-----
-----
Used-Bytes Used-Percentage Free-BytesFree-Percentage
71675904 55% 57483264 45%
-----
-----
-----
IPv4-ADDR IPv6-ADDR MAC Uptime Model MAX-APs
192.168.11.100NULL 00:13:11:01:01:01 12d 1h 29mZD111212
-----
-----
-----
Number-of-APs Number-of-ClientsNumber-of-Rogues Name
2 10ruckus
-----
-----
-----
Usage of 1 hr|Usage of 24 hr
Max-Concurrent-Users TX-BytesRogues | Max-Concurrent-Users TX-
BytesRogues
12.39M 02297.64M 2
```

```
-----  
-----  
Used-Bytes Used-Percentage Free-BytesFree-Percentage  
71675904 55% 57483264 45%  
-----  
-----  
-----
```



# Configuring Controller Settings

## 3

In this chapter:

- Configuration Commands Overview
- General Config Commands
- Configure Context Show Commands
- Configure Location Services Commands
- Configure AAA Server Commands
- Configure DHCP Server Commands
- Configure Admin Commands
- Configure Access Points Commands
- Configure AP Policy Commands
- Configure AP Group Commands
- Configure System Commands
- Configure WLAN Settings Commands
- Configure User Commands
- Configure Hotspot Commands
- Configure Mesh Commands
- Configure Alarm Commands
- Configure Services Commands
- Configure WIPS Commands
- Configure mDNS (Bonjour) Commands

# Configuration Commands Overview

This section describes the commands that you can use to configure ZoneDirector via the `config` context. From the privileged commands context, type `config` to enter the configuration context. To show a list of commands available from within the config context, type `help` or `?`.

## General Config Commands

The following section describes general configuration commands can be executed from within the config context. To save your configuration changes and exit the config context, use the `end` or `exit` command. To discard your changes and exit the config context, use the `abort` or `quit` command.

Some sub-contexts within the config context do not allow the use of the `abort` or `quit` commands; you must use save your changes and exit the sub-context. Many commands offer a corresponding “no” command to undo your configuration changes (for example, use “`no wlan`” to delete a WLAN).

### help

Shows available commands.

### history

Shows a list of previously run commands.

### abort

Exits the config context without saving changes. Some contexts do not allow abort, you must save your changes to exit the context (`end` or `exit`).

### end

Saves changes, and then exits the config context.

### exit

Saves changes, and then exits the config context.

### quit

Exits the config context without saving changes. Some contexts do not allow quit, you must save your changes to exit the context (`end` or `exit`).

## Configure Context Show Commands

Use the following `show` commands to display configured settings within the config context.

### **show aaa**

Displays a list of available AAA servers.

### **show dhcp**

Displays a list of available DHCP servers.

### **show admin**

Displays information about the administrator settings.

### **show mgmt-acl**

Displays a list of all management access controls.

### **show mgmt-acl-ipv6**

Displays a list of IPv6 management access controls.

### **show static-route**

Displays a list of all static route entries.

### **show static-route-ipv6**

Shows the static route for IPv6.

### **show ap**

Displays a list of all approved devices.

### **show l2acl**

Displays a list of L2 Access Control Lists.

### **show l3acl**

Displays a list of L3/L4/IP ACL.

**show whitelist**

Displays a list of client isolation white lists.

**show l3acl-ipv6**

Displays a list of L3/L4/IPV6 ACL.

**show prece**

Displays a list of Precedence Policies.

**show dvcpcy**

Displays a list of Device Policies.

**show app-denial-policy**

Displays the application denial policy settings.

**show user-defined-app**

Displays the user defined application settings.

**show app-port-mapping**

Displays the application category mapping settings.

**show load-balancing**

Displays information about Load balancing.

**show wlan**

Displays a list of all WLAN services (Names).

**show wlan-group**

Displays a list of existing WLAN groups.

**show role**

Displays a list of roles.

**show vlan-pool**

Displays a list of VLAN pools.

## **show user**

Displays a list of users.

## **show hotspot**

Displays a list of hotspot entries.

## **show guest-access-service**

To display a list of guest access services, use the following command:

```
show guest-access-service [all|name<WORD>]
```

## **show ap-group**

To display all or specified AP groups, use the following command:

```
show ap-group [all|name<WORD>]
```

## **show ap-policy**

Displays the ap policy settings.

## **show usb-software**

Displays USB Software Package information.

## **show location-services**

Displays a list of configured location services.

## **show mdnsproxyrule**

To display Mdnsproxy rules, use the following command:

```
show mdnsproxyrule <ID-From> <ID-to>
```

## **show mdnsproxy**

To display Mdnsproxy status, use the following command:

```
show mdnsproxy <ID-From> <ID-to>
```

## **show bonjour-policy**

To display Bonjour policy rules, use the following command:

```
show bonjour-policy <name>
```

# Configure Location Services Commands

This section describes the commands that you can use to configure Location Service entries on the controller. The following commands can be executed from within the `config-location-services` context. To show a list of commands available from within the `aaa` context, type `help` or `?`.

## location-services

To create or modify a location server, use the following command:

```
location-services <WORD>
```

### Syntax Description

<code>location-services &lt;WORD&gt;</code>	Creates a new location server or modifies an existing location server.
<code>abort</code>	Exits the <code>config-location-services</code> context without saving changes.
<code>end</code>	Saves changes, and then exits the <code>config-location-services</code> context.
<code>exit</code>	Saves changes, and then exits the <code>config-location-services</code> context.
<code>quit</code>	Exits the <code>config-location-services</code> context without saving changes.
<code>fqdn &lt;WORD&gt;</code>	Sets the location server FQDN.
<code>port &lt;PORT-NUM&gt;</code>	Sets the location server port.
<code>password &lt;WORD&gt;</code>	Sets the location server preshared key.
<code>show</code>	Displays configured location services for all venues.

### Example

```
ruckus(config)# location-services locationserver1
```

The location venue 'locationserver1' has been created. To save it, type 'end' or 'exit'.

```
ruckus(config-location-services)# fqdn ruckuslbs.ruckuswireless.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-location-services)# password secret1234
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-location-services)# show
Venue:
  ID:
  :
  Status = Disabled
  Venue Name = locationserver1
  Location Server FQDN = ruckuslbs.ruckuswireless.com
  Location Server Port = 8883
  Location Server PSK = secret1234
```

```
ruckus(config-location-services)# end
```

The location venue 'locationserver1' has been updated and saved. Your changes have been saved.

```
ruckus(config)#
```

## no location-services

To delete a location server from the list of location servers, use the following command:

```
no location-services <WORD>
```

# Configure AAA Server Commands

This section describes the commands that you can use to configure AAA server entries on the controller. The following commands can be executed from within the `config-aaa` context. To show a list of commands available from within the context, type `help` or `?`.

## aaa

Use the following command to configure an AAA server entry and enter the `config-aaa` context:

```
aaa <WORD>
```

**Syntax Description**

abort	Exits the config-aaa context without saving changes.
end	Saves changes, and then exits the config-aaa context.
exit	Saves changes, and then exits the config-aaa context.
quit	Exits the config-aaa context without saving changes.
name <WORD>	Sets the AAA server name.
show	Displays a list of available AAA servers.
type	Sets the type of AAA server.
type ad	Sets the AAA server type to 'Active Directory'.
type ldap	Sets the AAA server type to 'LDAP'.
type radius-auth	Sets the AAA server type to 'RADIUS'.
type tacplus-auth	Sets the AAA server type to 'TACPLUS'.
type radius-acct	Sets the AAA server type to 'RADIUS Accounting'.
radius-encryption	Sets the AAA server encryption type.
radius-encryption tls	Sets the AAA server encryption type to 'TLS'.
auth-method pap	Sets the authentication method to PAP.
auth-method chap	Sets the authentication method to CHAP.
ip-addr <IP-ADDR>	Sets the AAA server's IP/IPv6 address.
port <PORT-NUM>	Sets the AAA server's port.
tacplus-service <WORD>	Sets TACPLUS service name with length (1-64 bytes).
domain-name <WORD>	Sets the windows/base domain name.
no radius-encryption	Disables the AAA server encryption.
no ad-global-catalog	Disables global catalog support.
no grp-search	Disables group attribute lookup support.
no encryption-TLS	Disable the TLS Encryption
no backup	Disables the backup function.
ad-global-catalog	Enables global catalog support.
grp-search	Enables group attribute lookup support.



admin-dn <WORD>	Sets the admin domain name.
admin-password <WORD>	Sets the admin password.
key-attribute <WORD>	Sets the LDAP key attribute.
search-filter <WORD>	Sets the LDAP search filter.
radius-secret <WORD>	Sets the AAA server's shared secret.
tacplus-secret <WORD>	Sets the TACPLUS server's shared secret.
encryption-TLS	Enables the TLS Encryption
backup	Enables the backup function.
backup-ip-addr <IP- ADDR>	Sets the backup AAA server's IP/IPv6 address.
backup-port <PORT- NUM>	Sets the backup AAA server's port.
backup-radius-secret <WORD>	Sets the backup AAA server's shared secret.
request-timeout <NUMBER>	Sets the failover request timeout (2~20 seconds).
retry-count <NUMBER>	Sets the failover retry count (2~10 times).
consecutive-drop- packet <NUMBER>	Sets the number of consecutive dropped packet (range:1~10 , default is 1).
reconnect-primary- interval <NUMBER>	Sets the failover re-connect to primary interval (1~86400 minutes).

### **Example**

```
ruckus(config)# aaa activedir
```

The AAA server 'activedir' has been created. To save the AAA server, type 'end' or 'exit'.

```
ruckus(config-aaa)# type ad
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-aaa)# ip-addr 192.168.10.40
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-aaa) # show
AAA:
  ID:
    :
      Name= activedir
      Type= Active Directory
      IP Address= 192.168.10.40
      Port= 389
      Windows Domain Name=
      Global Catalog= Disabled
      Admin DN=
      Admin Password=
      Group Search= Enabled
      encryption-TLS = Disabled
```

```
ruckus(config-aaa) # end
The AAA server 'activedir' has been updated and saved.
Your changes have been saved.
ruckus(config) #
```

## Configure DHCP Server Commands

This section describes the commands that you can use to configure DHCP server entries on the controller. These DHCP server entries are used by the DHCP Relay feature, if enabled for a tunneled WLAN. The following commands can be executed from within the `config-dhcp` context.

### dhcp

Use the `dhcp` command from within the `config` context to create or edit a DHCP server entry.

```
dhcp <WORD>
```

### Syntax Description

<code>dhcp</code>	Configure the DHCP server settings
<code>&lt;WORD&gt;</code>	Name of the DHCP server entry

## Defaults

none

## Example

```
ruckus(config)# dhcp dhcp_server_2
```

The DHCP server 'dhcp\_server\_2' has been created. To save the DHCP server, type 'end' or 'exit'.

```
ruckus(config-dhcp)# first 192.168.11.99
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-dhcp)# show
```

DHCP servers for DHCP relay agent:

```
ID:
:
  Name= dhcp_server_2
  Description=
  IP Address= 192.168.11.99
```

```
ruckus(config-dhcp)# end
```

The DHCP server 'dhcp\_server\_2' has been updated and saved.

Your changes have been saved.

```
ruckus(config)# show dhcp
```

DHCP servers for DHCP relay agent:

```
ID:
1:
  Name= DHCP Server 1
  Description=
  IP Address= 192.168.11.1
  IP Address=

2:
  Name= dhcp_server_2
  Description=
  IP Address= 192.168.11.99
  IP Address=
```

```
ruckus(config)#
```

## no dhcp

Use the no dhcp command to delete a DHCP server entry.

```
no dhcp <WORD>
```

### *Example*

```
ruckus(config)# no dhcp dhcp_server_2  
The DHCP server 'dhcp_server_2' has been deleted.  
ruckus(config)#
```

## show

Displays a list of available DHCP servers.

```
show
```

## name

Sets the DHCP server name.

```
name <WORD>
```

## description

Sets the DHCP server description.

```
description <WORD>
```

## first

Sets the DHCP server's first IP address.

```
first <IP-ADDR>
```

## second

Sets the DHCP server's second IP address.

```
second <IP-ADDR>
```

## no second

Deletes the DHCP server's second IP address.

```
no second <IP-ADDR>
```

# Configure Admin Commands

Use the `admin` commands to enter the `config-admin` context to set the admin user name, password and admin authentication server settings.

## admin

To enter the `config-admin` context and configure administrator preference, use the following command:

```
admin
```

### Example

```
ruckus(config)# admin  
ruckus(config-admin)
```

## name

To set the administrator user name, use the following command:

```
name <WORD>
```

### Syntax Description

---

name	Configure the admin name setting
<WORD>	Set the admin name to this name

---

### Defaults

```
admin
```

### Example

```
ruckus(config)# admin  
ruckus(config-admin)# name admin  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-admin)# end  
The administrator preferences have been updated.  
Your changes have been saved.  
ruckus(config)#
```

## name password

To set the admin name and password at the same time, use the following command:

```
name <WORD> password <WORD>
```

### Syntax Description

name	Configure the admin name setting
<WORD>	Set the admin name to this name
password	Configure the admin password
<WORD>	Set the admin password to this password

### Defaults

admin

### Example

```
ruckus(config)# admin
```

```
ruckus(config-admin)# name admin password admin
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-admin)# end
```

The administrator preferences have been updated.

Your changes have been saved.

```
ruckus(config)#
```

## Admin Authentication Commands

Use the `auth-server` commands to set the administrator authentication options with an external authentication server.

### auth-server

To enable administrator authentication with a remote server and set the authentication server, use the following command:

```
auth-server <WORD>
```

## ***Syntax Description***

---

<code>auth-server</code>	Admin authentication with an external server
<code>&lt;WORD&gt;</code>	Set the authentication server to this server

---

## ***Defaults***

None.

## ***Example***

```
ruckus(config-admin)# auth-server radius
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-admin)#
```

## **no auth-server**

To disable administrator authentication with a remote server, use the following command:

```
no auth-server
```

## ***Syntax Description***

---

<code>no auth-server</code>	Disable admin authentication with an external server
-----------------------------	--

---

## ***Defaults***

None.

## ***Example***

```
ruckus(config-admin)# no auth-server
```

The command was executed successfully.

## **auth-server with-fallback**

To enable fallback authentication (for use when the remote server is unavailable), use the following command:

```
auth-server <WORD> with-fallback
```

### ***Syntax Description***

---

<code>auth-server</code>	Admin authentication with an external server
<code>&lt;WORD&gt;</code>	Set the auth-server to this server
<code>with-fallback</code>	Enable fallback authentication if the remote authentication server is unavailable

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-admin)# auth-server radius with-fallback
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-admin)# show
Administrator Name/Password:
Name= admin
Password= admin
Authenticate:
Mode= Authenticate with authentication server 'radius'
Fallback= Enabled

ruckus(config-admin)#
```

## **Configure Access Points Commands**

The following commands can be used from within the `config-ap` context to configure a specific Access Point.

### **ap**

To enter the `config-ap` context, enter the following command:

```
ap <MAC>
```



## Syntax Description

---

ap	Access Point
<MAC>	MAC address of the access point for configuration

---

## Defaults

None.

## Example

```
ruckus(config)# ap 04:4f:aa:0c:b1:00
```

The AP '04:4f:aa:0c:b1:00' has been loaded. To save the AP, type 'end' or 'exit' .

```
ruckus(config-ap)#
```

## no ap

To delete an AP from the list of approved devices, use the following command:

```
no ap <MAC>
```

## Syntax Description

---

no ap	Delete Access Point
<MAC>	MAC address of the access point

---

## Defaults

None.

## Example

```
ruckus(config)# no ap 04:4f:aa:0c:b1:00
```

The AP '04:4f:aa:0c:b1:00' has been deleted.

```
ruckus(config)#
```

## devname

To set the device name, use the following command:

devname <WORD>

## **Syntax Description**

---

devname	Device name
<WORD>	Set the device name to this name

---

## **Defaults**

None.

## **Example**

```
ruckus(config)# ap 04:4f:aa:0c:b1:00
```

The AP '04:4f:aa:0c:b1:00' has been loaded. To save the AP, type 'end' or 'exit'.

```
ruckus(config-ap)# devname 7962
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)# end
```

The device information has been updated.

Your changes have been saved.

```
ruckus(config)#
```

## **no devname**

To delete the device's name, use the following command:

```
no devname
```

## **bonjour-gateway**

To bind a bonjour gateway policy to this AP, use the following command:

```
bonjour-gateway <WORD>
```

## **Example**

```
ruckus(config-ap)# bonjour-gateway bonjour1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## **no bonjour-gateway**

To unbind a bonjour gateway policy, use the following command:

```
no bonjour-gateway
```

### **Example**

```
ruckus(config-ap)# no bonjour-gateway
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## **description**

To set the device description, use the following command:

```
description <WORD>
```

### **Syntax Description**

---

description	Device description
<WORD>	Set the device description to this text

---

### **Defaults**

None.

### **Example**

```
ruckus(config-ap-00:13:92:00:33:1C)# description this-is-the-device-description
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## **no description**

To delete the device's description, use the following command:

```
no description
```

## gps

To set the device GPS coordinates, use the following command:

```
gps <GPS-COORDINATE>
```

### Syntax Description

gps	Set the device GPS coordinates
<GPS-COORDINATE>	Enter the device's GPS coordinates for the latitude and longitude. Use a comma (,) to separate the latitude and longitude. The first coordinate is for the latitude. The second coordinate is for the longitude. Ex. A,B or -37,38.

### Defaults

None.

### Example

```
ruckus(config-ap)# gps 37.3,-122
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## no gps

To delete the device's GPS coordinates, use the following command:

```
no gps
```

## location

To set the device location, use the following command:

```
location <WORD>
```

### Syntax Description

location	Device location
<WORD>	Set the device location to this address

### Defaults

None.

### **Example**

```
ruckus(config-ap)# location sunnyvale-office
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

### **no location**

To delete the device's location, use the following command:

```
no location
```

### **group**

To set the AP group for this AP, use the following command:

```
group [name <WORD>] | system-default]
```

### **Syntax Description**

group	Set the AP group that this AP is a member of
name	Set the AP to be a member of the named AP group
<WORD>	The name of the AP group
system-default	Set the AP as a member of the system default AP group

### **Defaults**

```
system-default
```

### **Example**

```
ruckus(config-ap)# group system-default
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## ip

To set the AP's IPv4 address, use the following command from within the config-ap context:

```
ip [enable|disable] addr <IP-ADDR> <NET-MASK> name-server
<DNS-ADDR> mode [dhcp|static|keep]
```

### Syntax Description

ip	Set the AP's IPv4 addressing
enable	Enable IPv4 addressing
disable	Disable IPv4 addressing
addr	Set the AP's IPv4 address
<IP-ADDR>	The IPv4 address
<NET-MASK>	The IPv4 netmask
name-server	Set the device's DNS servers. Use a space ( ) to separate primary and secondary DNS servers
<DNS-ADDR>	The IP address of the DNS server
mode	Set the device's IP addressing mode (DHCP, static or "keep AP's setting")
dhcp	Set the device's IP address mode to DHCP
static	Set the device's IP address mode to static
keep	Set the device to use its current network settings

### Defaults

none

### Example

```
ruckus(config-ap)# ip enable mode dhcp
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## ipv6

To set the AP's IPv6 address, use the following command from within the config-ap context:

```
ipv6 [enable] addr <IPv6-ADDR> <IPv6-PREFIX-LENGTH> name-
server <DNS-ADDR> mode [auto|manual|keep]
```

### Syntax Description

ipv6	Set the AP's IPv6 addressing
enable	Enable IPv6 addressing
addr	Set the AP's IPv6 address
<IPv6-ADDR>	The IPv6 address
<IPv6-PREFIX-LENGTH>	The IPv6 prefix length. Use a space ( ) to separate the IPv6 address and prefix length
name-server	Set the device's DNS servers. Use a space ( ) to separate primary and secondary DNS servers
<DNS-ADDR>[<DNS-ADDR>]	The IP address of the DNS server
mode	Set the device's IP addressing mode (auto, manual or "keep AP's setting")
auto	Set the device's IPv6 address mode to auto
manual	Set the device's IPv6 address mode to manual
keep	Set the device to use its current network settings

### Defaults

none

### Example

```
ruckus(config-ap)# ipv6 enable mode auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)#
```

## no ipv6

To disable the AP's IPv6 mode, use the following command:

```
no ipv6
```

## Radio 2.4/5 GHz Commands

Use the `radio 2.4` or `radio 5` commands to configure the 2.4/5 GHz radio settings independently.

### radio

Use the `radio` command from within the `config-ap` context to configure the 2.4GHz or 5GHz radios independently.

```
radio [2.4|5] <arguments>
```

### Syntax Description

2.4	Configure the 2.4 GHz radio
5	Configure the 5 GHz radio
channelization [auto]<NUMBER>	Set channel width to 20 MHz, 40 MHz or Auto
channel [auto]<NUMBER>	Set channel to Auto or manually set channel
tx-power [auto full min num <1-10>]	Set transmit power to auto, full, min, or a number (-1dB~-10dB)
admission-control <VALUE>	Set the radio to use the specified call admission control airtime usage limit (%)
spectralink-compatibility [enable disable]	Enable SpectraLink Compatibility on the specified radio (set DTIM=2, minrate=5.5Mbps and enable RTS-CTS protection mode)
channel-range <NUMBER-LIST>	Set the allowed list of channels for the specified radio
wlan-group <WORD>	Set the AP radio as a member of a WLAN group
wlan-service [enable disable]	Enable WLAN service on this radio



---

extant-gain <NUMBER>	Set external antenna gain (on APs that support external antennas) (dBi)
-------------------------	---

---

### **Defaults**

channelization: Auto

channel: Auto

wlan-group: Default

wlan-service: Enabled

tx-power: Auto

admission-control: Disabled

spectralink-compatibility: Disabled

### **Example**

```
ruckus(config-ap)# radio 2.4 channelization auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)# radio 2.4 channel auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)# radio 2.4 wlan-group Default
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)# radio 2.4 wlan-service
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)# radio 2.4 tx-power auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap)# end
```

The device information has been updated.

Your changes have been saved.

```
ruckus(config)#
```

### **no radio**

Use the `no radio 2.4` or `no radio 5` commands from within the `config-ap` context to disable AP group overrides for the 2.4GHz or 5GHz radio settings.

```
no radio [2.4|5] <arguments>
```

## Syntax Description

no radio	Disable override of 2.4/5GHz radio settings
2.4	Disable 2.4GHz radio override settings
5	Disable 5GHz radio override settings
wlan-service	Disable override of WLAN service settings
channel-range-override	Disables override of channel range settings
channel-override	Disables override of channel settings
channelization-override	Disables override of 5GHz channelization settings
tx-power-override	Disables override of Tx power
wlan-group-override	Disables override of WLAN group settings
admission-control	Disables call admission control on the radio
admission-control-override	Disables override of call admission control settings
spectralink-compatibility-override	Disables the override of the Spectralink Compatibility settings
wlan-service	Disables WLAN service for the radio
channel-range-override	Disables override of channel range settings

## Example

```
ruckus(config-ap) # no radio 2.4 tx-power-override
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap) #
```

## mesh mode

Use the `mesh mode` command from within the `config-ap` context to configure the AP's mesh mode settings.

```
mesh mode [auto|root-ap|mesh-ap|disable]
```

## ***Syntax Description***

<code>mesh mode</code>	Configure the AP's mesh mode
<code>auto</code>	Set mesh mode to Auto
<code>root-ap</code>	Configure AP as a Root AP
<code>mesh-ap</code>	Configure AP as a Mesh AP
<code>disable</code>	Disable mesh

## ***Defaults***

Auto.

## ***Example***

```
ruckus(config-ap) # mesh mode auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap) #
```

## **mesh uplink-selection**

Use the `mesh uplink-selection` command from within the `config-ap` context to configure the AP's mesh uplink selection settings.

```
mesh uplink-selection [auto|manual] <add-mac>|<del-mac>  
<MAC>
```

## ***Syntax Description***

<code>mesh uplink-selection</code>	Configure the AP's mesh uplink selection mode
<code>auto</code>	Set mesh uplink selection to Auto
<code>manual</code>	Set mesh uplink selection to manual
<code>add-mac</code>	Add a manual uplink selection AP
<code>del-mac</code>	Delete a manual uplink selection AP
<code>&lt;MAC&gt;</code>	The MAC address of the uplink AP

## Defaults

Auto.

## Example

```
ruckus(config-ap) # mesh uplink-selection manual add-mac
00:24:82:3f:14:60
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap) #
```

## Example

```
ruckus(config-ap) # mesh uplink-selection auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap) #
```

## status-leds

To enable or disable the AP's status LEDs, use the following command:

```
status-leds [enable|disable]
```

## Defaults

Enabled.

## Syntax Description

status-leds	Configure status LEDs
enable	Override group config, enable status LEDs
disable	Override group config, disable status LEDs

## Example

```
ruckus(config-ap) # status-leds disable
```

```
ruckus(config-ap) #
```

## no status-leds-override

To disable override of status LEDs for this AP, use the following command:

```
no status-leds-override
```

## usb-port

To disable the override the group configuration and enable/disable the USB port for this AP, use the following command:

```
usb-port [enable|disable]
```

## no usb-port-override

To disable the override of the USB port for the specified AP model, use the following command:

```
no usb-port-override
```

## poe-out

To enable or disable the AP's PoE Out port, use the following command:

```
poe-out [enable|disable]
```

## Defaults

Disabled.

## Syntax Description

<code>poe-out</code>	Configure PoE Out port
<code>enable</code>	Override group config, enable PoE Out port
<code>disable</code>	Override group config, disable PoE Out port

## Example

```
ruckus(config-ap)# poe-out enable  
ruckus(config-ap)#
```

## no poe-out-override

To disable override of the PoE out port settings, use the following command:

```
no poe-out-override
```

## **no usb-software-override**

To disable the override of the AP USB software package, use the following command:

```
no usb-software-override
```

## **external-antenna**

To configure the AP's external antenna settings, use the following command:

```
external-antenna [2.4G|5G] [enable|disable] [gain  
<NUMBER>] cable-loss <NUMBER> [2-antennas|3-antennas]
```

### ***Syntax Description***

2.4G	Configure external 2.4GHz antenna
5G	Configure external 5GHz antenna
enable disable	Enable/disable external antenna
gain	Set external antenna gain for 2.4/5GHz radio
cable-loss <NUMBER>	Enter the external antenna loss (0-90 dB)
2-antennas	Select two external antennas for the specified radio
3-antennas	Select three external antennas for the specified radio

### ***Defaults***

Varies by AP model.

## **no external-antenna-override**

To disable the external antenna override settings, use the following command:

```
no external-antenna-override
```

## **spectra-analysis 2.4GHz**

To enable or disable the spectrum analysis feature for this radio, use the following command:

```
spectra-analysis 2.4GHz [enable|disable]
```

## **spectra-analysis 5GHz**

To enable or disable the spectrum analysis feature for this radio, use the following command:

```
spectra-analysis 5GHz [enable|disable]
```

## **internal-heater**

To enable or disable the AP's internal heater, use the following command:

```
internal-heater [enable|disable]
```

### ***Defaults***

Disabled.

### ***Syntax Description***

---

<code>internal-heater</code>	Configure internal heater
<code>enable</code>	Override group config, enable internal heater
<code>disable</code>	Override group config, disable internal heater

---

### ***Example***

```
ruckus(config-ap) # internal-heater enable  
ruckus(config-ap) #
```

## **no internal-heater-override**

To disable override of the internal heater for this AP, use the following command:

```
no internal-heater-override
```

## **cband-channels**

To enable or disable the 5.8 GHz C-band channels, use the following command:

```
cband-channels [enable|disable]
```

### ***Defaults***

Disabled.

## Syntax Description

---

<code>cband-channels</code>	Configure C-band channels
<code>enable</code>	Override group config, enable C-band channels
<code>disable</code>	Override group config, disable C-band channels

---

### Example

```
ruckus(config-ap) # cband-channels enable
ruckus(config-ap) #
```

### no cband-channels-override

To disable override of the 5.8 GHz channels, use the following command:

```
no cband-channels-override
```

### usb-software

To set the AP USB software package vendor ID (VID) and product ID (PID), and version, use the following command:

```
usb-software <VID-PID-VERSION>
```

### no usb-software

To delete a USB software package from the list of USB software packages, use the following command:

```
no usb-software
```

### ipmode

To set the AP's IP mode, use the following command:

```
ipmode <WORD>
```

### Defaults

Dual-stack IPv4/IPv6 mode



## ***Syntax Description***

---

<code>ipmode</code>	Configure IP addressing mode
<code>ipv4</code>	Set to IPv4 only mode
<code>ipv6</code>	Set to IPv6 only mode
<code>dual</code>	Set to dual-stack IPv4/IPv6 mode

---

### ***Example***

```
ruckus(config-ap)# ipmode dual  
ruckus(config-ap)#
```

### **no ipmode-override**

To disable override of the IP mode, use the following command:

```
no ipmode-override
```

### **radio-band**

To set the radio band of the AP, use the following command:

```
radio-band <WORD>
```

This command is available only on APs that support band switching between 2.4GHz and 5GHz radio band modes.

## ***Syntax Description***

---

<code>radio-band</code>	Configure radio band mode
<code>&lt;WORD&gt;</code>	Set to 2.4 or 5 GHz radio mode

---

### ***Example***

```
ruckus(config-ap) # radio-band 5  
Your changes have been saved.  
ruckus(config-ap) #
```

### **no radio-band-override**

To disable the AP radio band override, use the following command:

```
no radio-band-override
```

## **venue-name**

To set the venue name of the AP, use the following command:

```
venue-name [language] <WORD>
```

### ***Syntax Description***

venue-name	Set the venue name for the AP
[language]	Set the language of the venue name. Valid languages are: English, Chinese, Czech, Danish, Dutch, French, German, Japanese, Spanish, Swedish, Turkish)
<WORD>	Set the venue name to the name specified

### ***Example***

```
ruckus(config-ap) # venue-name english venue1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap) #
```

## **no venue-name**

To remove a venue name entry, use the following command:

```
no venue-name [language]
```

### ***Example***

```
ruckus(config-ap) # no venue-name english
```

The entry 'English' has been removed. To save the changes, type 'end' or 'exit'.

```
ruckus(config-ap) #
```

## **lldp**

To enable, disable or configure the AP's Link Layer Discover Protocol settings, use the following lldp commands from within the config-ap context.

## ***Syntax Description***

lldp	Configure LLDP settings.
enable	Enable LLDP with current settings.
disable	Disable LLDP with current settings.
interval <NUMBER>	Set packet transmit interval in second(s).
holdtime <NUMBER>	Set amount of time receiving device should retain the information.
ifname eth <NUMBER>	Enter the AP port number.
mgmt enable	Enable LLDP management IP address of the AP.
mgmt disable	Disable LLDP management IP address of the AP.

### ***Example***

```
ruckus(config-ap) # lldp enable  
ruckus(config-ap) #
```

### **no lldp-override**

To disable the AP's LLDP override settings (use parent settings), use the following command:

```
no lldp-override
```

### ***Example***

```
ruckus(config-ap) # no lldp-override  
ruckus(config-ap) #
```

### **show**

To display the AP's current configuration settings, use the following command:

```
show
```

### ***Example***

```
ruckus(config)# ap c4:10:8a:1f:d1:f0
```

The AP 'c4:10:8a:1f:d1:f0' has been loaded. To save the AP, type 'end' or 'exit'.

```
ruckus(config-ap)# show
AP:
ID:
  1:
    MAC Address= c4:10:8a:1f:d1:f0
    Model= zf7982
    Approved= Yes
    Device Name= 7982
    Description=
    Location=
    GPS=
    CERT = Normal
    Group Name= System Default
    Channel Range:
      A/N= 36,40,44,48,149,153,157,161 (Disallowed= )
      B/G/N= 1,2,3,4,5,6,7,8,9,10,11 (Disallowed= )
    Radio a/n:
      Channelization= Auto
      Channel= Auto
      WLAN Services enabled= Yes
      Tx. Power= Auto
      WLAN Group Name= Default
      Call Admission Control= OFF
      SpectraLink Compatibility= Disabled
    Radio b/g/n:
      Channelization= Auto
      Channel= Auto
      WLAN Services enabled= Yes
      Tx. Power= Auto
      WLAN Group Name= Default
      Call Admission Control= OFF
```

SpectraLink Compatibility= Disabled  
Override global ap-model port configuration= No

**Network Setting:**

Protocol mode= Use Parent Setting  
Device IP Settings= Keep AP's Setting  
IP Type= DHCP  
IP Address= 192.168.40.64  
Netmask= 255.255.255.0  
Gateway= 192.168.40.1  
Primary DNS Server= 192.168.40.1  
Secondary DNS Server=

Device IPv6 Settings= Keep AP's Setting  
IPv6 Type= Auto Configuration  
IPv6 Address= fc00::1  
IPv6 Prefix Length= 7  
IPv6 Gateway=  
IPv6 Primary DNS Server=  
IPv6 Secondary DNS Server=

**Mesh:**

Status= Enabled  
Mode= Auto

**Uplink:**

Status= Smart

**Venue Name List:****LAN Port:**

0:  
Interface= eth0  
Dot1x= None  
LogicalLink= Down  
PhysicalLink= Down

```
Label= 10/100/1000 PoE LAN1
1:
Interface= eth1
Dot1x= None
LogicalLink= Up
PhysicalLink= Up 1000Mbps full
Label= 10/100/1000 LAN2
```

```
ruckus(config-ap)#
```

## AP Port Setting Commands

To override AP group configuration settings and configure the AP's Ethernet ports individually, you must first enter the `config-ap-model` context from within the `config-ap` context.

### port-setting

Use the following command to enter the `config-ap-model` context and override AP group settings to configure AP ports individually:

```
port-setting
```

### Defaults

```
Enable LAN: Yes
LAN Type: trunk
Untag ID: 1
Members: 1-4094
Guest VLAN: Disabled
Dynamic VLAN: Disabled
802.1X: disabled
DHCP opt82: Disabled
Tunnel= Disabled
MLD Snooping: Disabled
IGMP Snooping: Enabled
```

## Syntax Description

port-setting	Configure AP port settings
lan <NUMBER> {Arguments}	Configure the AP LAN port
no lan <NUMBER>	Disable the AP LAN port
uplink <WORD>	Set the AP port to use the specified type (trunk, access or general)
untag <NUMBER>	Set the AP port to use the specified VLAN ID(1-4094)
member <NUMBER>	Set the AP port to use the specified members(1-4094)
opt82 [enabled disabled]	Enable the AP port DHCP Option 82 settings
tunnel [enabled disabled]	Enable the AP port tunnel settings
guest-vlan <NUMBER>	Set the AP port to use the specified guest VLAN ID(1-4094)
dvlan [disabled enabled]	Enable the AP port dynamic VLAN settings
no dot1x <authsvr> <acctsvr> <mac-auth-bypass>	Disable authentication server, accounting server, or MAC auth bypass for the AP's 802.1X settings
dot1x <authsvr> <acctsvr> <mac-auth-bypass>	Enable authentication server, accounting server, or MAC auth bypass for the AP's 802.1X settings
authsvr <WORD>	Enter the RADIUS server name
acctsvr <WORD>	Enter the RADIUS accounting server name
mac-auth-bypass	Enable MAC authentication bypass for the 802.1X-enabled port
dot1x supplicant [username password] <WORD>	Set the username/password for AP 802.1X supplicant
dot1x supplicant mac	Set the username and password to use AP MAC address for AP 802.1X supplicant

## **Example**

```
ruckus(config-ap) # port-setting
ruckus(config-ap-model) # lan 1 uplink trunk
ruckus(config-ap-model) # show
PORTS:
  LAN ID:
    1:
      Enable LAN = Yes
      LAN Type= trunk
      Untag ID= 1
      Members= 1-4094
      Guest VLAN=
      Enable Dynamic VLAN= Disabled
      802.1X= disabled
      DHCP opt82= Disabled
      Tunnel= Disabled
      MLD Snooping= Disabled
      IGMP Snooping= Enabled
    2:
      Enable LAN = Yes
      LAN Type= trunk
      Untag ID= 1
      Members= 1-4094
      Guest VLAN=
      Enable Dynamic VLAN= Disabled
      802.1X= disabled
      DHCP opt82= Disabled
      Tunnel= Disabled
      MLD Snooping= Disabled
      IGMP Snooping= Enabled
ruckus(config-ap-model) #
```

## **abort**

To exit the `port-setting` context without saving changes, use the `abort` command.

```
abort
```



## ***Syntax Description***

---

abort	Exit the context without saving changes
-------	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-ap-model)# abort
No changes have been saved.
ruckus(config-ap)#
```

## **end**

To save changes, and then exit the `port-setting` context, use the following command:

```
end
```

## ***Syntax Description***

---

end	Save changes, and then exit the context
-----	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-ap-model)# end
ruckus(config-ap)#
```

## **exit**

To save changes, and then exit the `config-ap-model` context, use the following command:

```
exit
```

## ***Syntax Description***

---

<code>exit</code>	Save changes, and then exit the context
-------------------	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-ap-model)# exit
ruckus(config-ap)#
```

## **quit**

To exit the `config-ap-model` context without saving changes, use the `quit` command.

```
quit
```

## ***Syntax Description***

---

<code>quit</code>	Exit the context without saving changes
-------------------	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-ap-model)# quit
No changes have been saved.
ruckus(config-ap)#
```

## **show**

To display the current port settings, use the following command:

```
show
```

## Syntax Description

---

show	Display the current port settings
------	-----------------------------------

---

### Defaults

None.

### Example

```
ruckus(config)# ap 04:4f:aa:0c:b1:00
ruckus(config-ap)# port-setting
ruckus(config-ap-model)# show
ruckus(config-ap-model)# show
PORTS:
  LAN ID:
    1:
      Enable LAN = Yes
      LAN Type= trunk
      Untag ID= 1
      Members= 1-4094
      Guest VLAN=
      Enable Dynamic VLAN= Disabled
      802.1X= disabled
      DHCP opt82= Disabled
      Tunnel= Disabled
      MLD Snooping= Disabled
      IGMP Snooping= Enabled
    2:
      Enable LAN = Yes
      LAN Type= trunk
      Untag ID= 1
      Members= 1-4094
      Guest VLAN=
      Enable Dynamic VLAN= Disabled
      802.1X= disabled
      DHCP opt82= Disabled
      Tunnel= Disabled
      MLD Snooping= Disabled
```

```

IGMP Snooping= Enabled
ruckus(config-ap-model) #

```

## lan

To enable the LAN port, use the following command:

```
lan <NUMBER>
```

### Syntax Description

lan	Enable the LAN port
<NUMBER>	Specify the LAN port to enable
uplink <WORD>	Sets the AP port to use the specified type(trunk,access or general).
untag <NUMBER>	Sets the AP port to use the specified VLAN ID(1-4094) or none.
member <NUMBER>	Sets the AP port to use the specified members(1-4094).
opt82	Sets the AP port DHCP Option 82.
tunnel	Sets the AP port tunnel.
guest-vlan <NUMBER>	Sets the AP port to use the specified guest VLAN ID(1-4094).
dvlan	Sets the AP port dynamic VLAN.
dot1x	Sets the AP port 802.1X.

### Defaults

```

Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
Guest VLAN=
Enable Dynamic VLAN= Disabled
802.1X= disabled
DHCP opt82= Disabled
Tunnel= Disabled

```

```
MLD Snooping= Disabled
```

```
IGMP Snooping= Enabled
```

### **Example**

```
ruckus(config-ap-model)# lan 1
```

```
ruckus(config-ap-model)#
```

### **no lan**

To disable the LAN port, use the following command:

```
no lan <NUMBER>
```

### **Syntax Description**

no lan	Disable the LAN port
<NUMBER>	Specify the LAN port to disable

### **Defaults**

None.

### **Example**

```
ruckus(config-ap-model)# no lan 1
```

```
ruckus(config-ap-model)#
```

### **lan uplink**

To sets the AP port type (Trunk, Access or General), use the following command:

```
lan <NUMBER> uplink <WORD>
```

### **Syntax Description**

lan uplink	Set the LAN port type
<NUMBER>	Specify the LAN port to configure

uplink	Set the port type to the specified type
<WORD>	LAN port type (Trunk port, Access port, General port)

### **Defaults**

For all APs other than 7025/7055: Trunk

For 7025/7055 LAN 5: Trunk

For 7025/7055 LAN 1-LAN 4: Access

### **Example**

```
ruckus(config-ap-model)# lan 1 uplink access
ruckus(config-ap-model)#
```

### **lan untag**

To set the LAN port untag VLAN ID (native VLAN, for Trunk ports), use the following command:

```
lan <NUMBER> untag <NUMBER>
```

### **Syntax Description**

lan untag	Set the LAN port untag VLAN ID
<NUMBER>	Specify the LAN port to configure
<NUMBER>	Set the untag VLAN ID (1~4094)

### **Defaults**

1

### **Example**

```
ruckus(config-ap-model)# lan 1 untag 1
ruckus(config-ap-model)#
```

## lan member

To set the LAN port VLAN membership (only General ports have configurable membership; Trunk ports are members of all VLANs, and Access port membership must be the same as the Untag VLAN), use the following command:

```
lan <NUMBER> member <NUMBER>
```

### Syntax Description

lan member	Set the LAN port VLAN membership
<NUMBER>	Specify the LAN port to configure
<NUMBER>	Set the VLAN membership (1~4094, range separated by hyphen, multiple VLANs separated by commas)

### Defaults

1

### Example

```
ruckus(config-ap-model)# lan 2 member 1-10,100,200
```

```
ruckus(config-ap-model)# show
```

PORTS:

LAN ID:

1:

```
Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
Guest VLAN=
Enable Dynamic VLAN= Disabled
802.1X= disabled
DHCP opt82= Disabled
Tunnel= Disabled
MLD Snooping= Disabled
IGMP Snooping= Enabled
```

2:

```
Enable LAN = Yes
```

```

LAN Type= general
Untag ID= 1
Members= 1-10,100,200
Guest VLAN=
Enable Dynamic VLAN= Disabled
802.1X= disabled
DHCP opt82= Disabled
Tunnel= Disabled
MLD Snooping= Disabled
IGMP Snooping= Enabled
ruckus(config-ap-model)#
    
```

## lan opt82

To enable or disable DHCP option 82 for a LAN port, use the following command:

```
lan <NUMBER> opt82 [enabled|disabled]
```

### *Syntax Description*

opt82	Enable or disable DHCP option 82
enabled	Enable option 82
disabled	Disable option 82

### *Defaults*

Disabled

### *Example*

```

ruckus(config-ap-model)# lan 1 opt82 enable
ruckus(config-ap-model)#
    
```

## lan tunnel

To enable or disable Ethernet port tunnel mode for the port, use the following command:

```
lan <NUMBER> tunnel [enabled|disabled]
```



## Syntax Description

---

tunnel	Enable or disable port tunnel mode
enabled	Enable tunnel mode
disabled	Disable tunnel mode

---

## Defaults

Disabled

## Example

```
ruckus(config-ap-model)# lan 1 tunnel enable
```

```
ruckus(config-ap-model)# show
```

PORTS:

LAN ID:

1:

```
Enable LAN = Yes  
LAN Type= trunk  
Untag ID= 1  
Members= 1-4094  
Guest VLAN=  
Enable Dynamic VLAN= Disabled  
802.1X= disabled  
DHCP opt82= Disabled  
Tunnel= Enabled  
MLD Snooping= Disabled  
IGMP Snooping= Enabled
```

2:

```
Enable LAN = Yes  
LAN Type= trunk  
Untag ID= 1  
Members= 1-4094  
Guest VLAN=  
Enable Dynamic VLAN= Disabled  
802.1X= disabled  
DHCP opt82= Disabled  
Tunnel= Disabled
```

```

MLD Snooping= Disabled
IGMP Snooping= Enabled
ruckus(config-ap-model) #

```

## lan guest-vlan

To set the AP port to use the specified Guest VLAN ID, use the following command:

```
lan <NUMBER> guest-vlan <NUMBER>
```

## lan dvlan enabled

To enable dynamic VLAN for the port, use the following command:

```
lan <NUMBER> dvlan enabled
```

## lan dvlan disabled

To disable dynamic VLAN for the port, use the following command:

```
lan <NUMBER> dvlan disabled
```

## lan dot1x

To configure 802.1X settings for a LAN port, use the following command:

```
lan <NUMBER> dot1x [disable|supplicant|auth-port-
based|auth-mac-based]
```

## Syntax Description

lan dot1x	Configure 802.1X settings for this port
<NUMBER>	LAN port number to configure
disabled	Disable 802.1X
supplicant	Configure this LAN port as an 802.1X supplicant
supplicant username <WORD>	Set the username for AP 802.1X supplicant
supplicant password <WORD>	Set the password for AP 802.1X supplicant
supplicant mac	Set the username and password to use AP MAC address for AP 802.1X supplicant
auth-port-based	Configure this LAN port as an 802.1X authenticator (port-based)

---

auth-mac-based	Configure this LAN port as an 802.1X authenticator (MAC-based)
----------------	--

---

### **Defaults**

Disabled

### **Example**

```
ruckus(config-ap-model)# lan 1 dot1x supplicant
ruckus(config-ap-model)#
```

### **dot1x authsvr**

To configure the 802.1X authentication server for the AP, use the following command:

```
dot1x authsvr <WORD>
```

### **Syntax Description**

---

dot1x authsvr	Configure 802.1X authentication server
<WORD>	Name of AAA server

---

### **Defaults**

None

### **Example**

```
ruckus(config-ap-model)# dot1x authsvr radius
ruckus(config-ap-model)#
```

### **dot1x acctsvr**

To configure the 802.1X accounting server for the AP, use the following command:

```
dot1x acctsvr <WORD>
```

## ***Syntax Description***

---

dot1x acctsvr	Configure 802.1X accounting server
<WORD>	Name of AAA server

---

## ***Defaults***

None

## ***Example***

```
ruckus(config-ap-model) # dot1x acctsvr radius-acct
ruckus(config-ap-model) #
```

## **dot1x mac-auth-bypass**

To configure 802.1X MAC authentication bypass, use the following command:

```
dot1x mac-auth-bypass
```

## ***Syntax Description***

---

dot1x mac-auth-bypass	Enable 802.1X MAC authentication bypass
-----------------------	---

---

## ***Defaults***

Disabled

## ***Example***

```
ruckus(config-ap-model) # dot1x mac-auth-bypass
ruckus(config-ap-model) #
```

## **dot1x supplicant username**

To configure 802.1X supplicant user name, use the following command:

```
dot1x supplicant username <WORD>
```

## Syntax Description

---

dot1x supplicant username	Configure 802.1X supplicant user name
<WORD>	Set the 802.1X supplicant user name

---

## Defaults

None

## Example

```
ruckus(config-ap-model) # dot1x supplicant username johndoe  
ruckus(config-ap-model) #
```

## dot1x supplicant password

To configure 802.1X supplicant password, use the following command:

```
dot1x supplicant password <WORD>
```

## Syntax Description

---

dot1x supplicant password	Configure 802.1X supplicant password
<WORD>	Set the 802.1X supplicant password

---

## Defaults

None

## Example

```
ruckus(config-ap-model) # dot1x supplicant password test123  
ruckus(config-ap-model) #
```

## dot1x supplicant mac

To set the 802.1X supplicant user name and password as the AP's MAC address, use the following command:

```
dot1x supplicant mac
```

## Syntax Description

---

dot1x supplicant mac	Set the supplicant user name and password as the AP's MAC address
----------------------	---

---

## Defaults

None

## Example

```
ruckus(config-ap-model) # dot1x supplicant mac
ruckus(config-ap-model) #
```

# Configure AP Policy Commands

Use the `ap-policy` commands to configure global AP policies such as automatic AP approval, limited ZD discovery, management VLAN, load balancing across APs and max clients per AP radio. To run these commands, you must first enter the `config-ap-policy` context.

## ap-policy

To enter the `ap-policy` context and configure global AP policies, enter the following command:

```
ap-policy
```

## Syntax Description

---

ap-policy	Enter config-ap-policy context and configure global AP policies
-----------	---

---

## Defaults

None.

## Example

```
ruckus(config)# ap-policy
ruckus(config-ap-policy)#
```

## show

To display the current device policy, use the following command:

```
show
```

## Syntax Description

---

show	Display the current AP policy settings
------	--

---

## Defaults

None.

## Example

```
ruckus(config-ap-policy)# show
  Automatically approve all join requests from APs= Enabled
  Limited ZD Discovery:
    Status= Disabled
  Management VLAN:
    Status= Keep AP's setting
  Balances the number of clients across adjacent APs= Disabled
  LWAPP message MTU= 1450
  Auto Recovery= 30 minutes
ruckus(config-ap-policy)#
```

## ap-management-vlan

To enable the AP management VLAN and set to either “keep AP’s setting” or to the specified VLAN ID, use the following command:

```
ap-management-vlan [keeping] <NUMBER>
```

## Syntax Description

---

<code>ap-management-vlan</code>	Enable and configure the global AP management VLAN
<code>keeping</code>	Sets management VLAN to “Keep AP’s setting”
<code>&lt;NUMBER&gt;</code>	Set management VLAN to the number specified

---

## Defaults

None.

## Example

```
ruckus(config-ap-policy) # ap-management-vlan keeping
The command was executed successfully.
ruckus(config-ap-policy) #
```

## no ap-management-vlan

To disable the AP management VLAN, use the following command:

```
no ap-management-vlan
```

## Syntax Description

---

<code>no ap-management-vlan</code>	Disable the AP management VLAN
------------------------------------	--------------------------------

---

## Defaults

None.

```
ruckus(config-ap-policy) # no ap-management-vlan
```

## Example

```
The command was executed successfully.
ruckus(config-ap-policy) #
```



## ap-auto-approve

To enable the automatic approval of join requests from devices, use the following command:

```
ap-auto-approve
```

### **Syntax Description**

---

ap-auto-approve	Enable the automatic approval of join requests from devices
-----------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-ap-policy) # ap-auto-approve
```

The AP automatically approve policy has been updated.

## no ap-auto-approve

To disable the automatic approval of join requests from devices, use the following command:

```
no ap-auto-approve
```

### **Syntax Description**

---

no ap-auto-approve	Disable the automatic approval of join requests from devices
--------------------	--

---

### **Defaults**

None.

### **Example**

```
ruckus(config-ap-policy) # no ap-auto-approve
```

The AP automatically approve policy has been updated.

```
ruckus(config-ap-policy) #
```

## limited-zd-discovery

To configure devices to connect to a specific ZoneDirector and to set the primary and secondary ZoneDirector's IP addresses, use the following command:

```
limited-zd-discovery <zd-addr|zd-ip> <PRIMARY>
<SECONDARY>
```

### Syntax Description

limited-zd-discovery	Configure devices to connect to a specific ZoneDirector
zd-addr	Set ZoneDirector's IP/IPv6/FQDN address
zd-ip	Set ZoneDirector's IP/IPv6 address
<PRIMARY>	Address of primary ZD
<SECONDARY>	Address of secondary ZD

### Defaults

Disabled.

### Example

```
ruckus(config-ap-policy)# limited-zd-discovery zd-addr
192.168.11.100 192.168.11.200
```

The Limited ZoneDirector discovery function has been updated.

```
ruckus(config-ap-policy)# show
```

```
Automatically approve all join requests from APs= Enabled
```

```
Limited ZD Discovery:
```

```
Status= Enabled
```

```
Primary ZoneDirector ADDR= 192.168.11.100
```

```
SecondaryZoneDirector ADDR= 192.168.11.200
```

```
Prefer Primary ZoneDirector = false
```

```
Management VLAN:
```

```
Status= Disabled
```

```
Balances the number of clients across adjacent APs= Disabled
```

```
Max. clients for 11BG radio= 100
```

```
Max. clients for 11N radio= 100
```

```
LWAPP message MTU= 1450  
ruckus(config-ap-policy) #
```

## **no limited-zd-discovery**

To disable limited ZD discovery, use the following command:

```
no limited-zd-discovery
```

### ***Syntax Description***

---

no limited-zd-	Disable limited ZD discovery
discovery	

---

### ***Defaults***

Disabled.

### ***Example***

```
ruckus(config-ap-policy) # no limited-zd-discovery  
The Limited ZoneDirector discovery function has been updated.  
ruckus(config-ap-policy) #
```

## **limited-zd-discovery prefer-primary-zd**

To force the AP to prefer the primary ZoneDirector when connected (and periodically attempt to reconnect to the primary ZD when disconnected from it), use the following command:

```
limited-zd-discovery prefer-primary-zd
```

### ***Example***

```
ruckus(config-ap-policy) # limited-zd-discovery prefer-primary-zd  
The Limited ZoneDirector discovery function has been updated.  
ruckus(config-ap-policy) #
```

## **no limited-zd-discovery prefer-primary-zd**

To disable the Limited ZD Discovery “prefer primary ZoneDirector” feature, use the following command:

```
no limited-zd-discovery prefer-primary-zd
```

## **limited-zd-discovery keep-ap-setting**

To disallow ZoneDirector modifying AP’s original primary/secondary ZD settings, use the following command:

```
limited-zd-discovery keep-ap-setting
```

### ***Example***

```
ruckus(config-ap-policy)# limited-zd-discovery keep-ap-setting  
The Limited ZoneDirector discovery function has been updated.  
ruckus(config-ap-policy)#
```

```
no limited-zd-discovery keep-ap-setting
```

To disable the Limited ZD Discovery “keep AP’s setting” feature, use the following command:

```
no limited-zd-discovery keep-ap-setting
```

## **auto-recovery**

To set the value of auto recovery time (minutes) for AP reboot if AP can't connect to ZoneDirector, use the following command:

```
auto-recovery <NUMBER>
```

### ***Defaults***

Enabled

30 minutes

### ***Example***

```
ruckus (config-ap-policy)# auto-recovery 30  
The AP auto recovery policy has been updated.  
ruckus(config-ap-policy)#
```

## no auto-recovery

To disable AP auto recovery, use the following command:

```
no auto-recovery
```

## vlan-qos

To configure the traffic class [Voice | Video | Data | Background] to the specific VLAN ID at the specific interface, use the following command:

```
vlan-qos <VID> <Traffic Class> <Interface Name>
```

## Syntax Description

vlan-qos	Configure VLAN QOS settings
<VID>	VLAN ID
<Traffic Class>	Specify traffic classification (voice, video, data, background)
<Interface Name>	Specify interface name

## Defaults

Disabled

## Example

```
ruckus(config-ap-policy)# vlan-qos 10 voice eth0
The VLAN QoS function has been updated.
ruckus(config-ap-policy)#
```

## no vlan-qos

To disable VLAN traffic class QoS for the specific interface or all VLANs, use the following command:

```
no vlan-qos <all|VID> <Interface Name>
```

## Syntax Description

no vlan-qos	Disable VLAN's QoS settings
<VID>	VLAN ID
<Interface Name>	Specify interface name

## Defaults

Disabled

## Example

```
ruckus(config-ap-policy)# no vlan-qos all eth0
The VLAN QoS function has been updated.
ruckus(config-ap-policy)#
```

## timeout

To configure recovering of the APs' original Primary/Secondary ZD address if the AP can't find the desired Primary/Secondary ZD after timeout(minutes), use the following command:

```
timeout <NUMBER>
```

## Syntax Description

timeout	Enter the timeout value (minutes) for recovering APs' original primary/secondary ZD IP.
<NUMBER>	Timeout value in minutes.

## Example

```
ruckus(config-ap-policy-move-ap)# timeout 60
Your changes have been saved.
ruckus(config-ap-policy-move-ap)#
```

## no timeout

To disable the timeout function for moving APs, use the following command:

```
no timeout
```

## import-aplist

To import an AP list from backup files on a TFTP server, use the following command:

```
import-aplist <IP-ADDR> <FILE-NAME>
```

## exit

Saves changes, and then exits the config-ap-policy-move-ap context.

## abort

Exits the config-ap-policy-move-ap context without saving changes.

## quit

Exits the config-ap-policy-move-ap context without saving changes.

## show

Displays the AP policy settings.

### **Example**

```
ruckus(config-ap-policy)# show
```

```
Automatically approve all join requests from APs= Enabled
```

```
Limited ZD Discovery:
```

```
Status= Disabled
```

```
Management VLAN:
```

```
Status= Keep AP's setting
```

```
Balances the number of clients across adjacent APs= Disabled
```

```
Auto Recovery= 30 minutes
```

```
ruckus(config-ap-policy)#
```

## Configure AP Group Commands

This section describes the commands that you can use to configure AP groups on the controller. The following commands can be executed from within the `config-apgrp` context. To show a list of commands available from within the context, type `help` or `?`.

## ap-group

To create a new AP group or configure an existing AP group and enter the `config-apgrp` context, enter the following command:

```
ap-group <WORD>
```

### Syntax Description

<code>ap-group</code>	Configure an AP group
<code>&lt;WORD&gt;</code>	Name of the AP group

### Defaults

“System Default”

### Example

```
ruckus(config)# ap-group "System Default"
```

The AP group entry 'System Default' has been loaded. To save the AP group, type 'end' or 'exit'.

```
ruckus(config-apgrp)#
```

## no ap-group

To delete an AP group from the list, enter the following command:

```
no ap-group <WORD>
```

### Syntax Description

<code>no ap-group</code>	Delete an AP group
<code>&lt;WORD&gt;</code>	Name of the AP group

### Defaults

None

### Example

```
ruckus(config)# no ap-group apgrp2
```



```
The AP Group 'apgrp2' has been removed.  
ruckus(config)#
```

## **exit**

Saves changes, and then exits the config-ap-group context.

## **abort**

Exits the config-ap-group context without saving changes.

## **quit**

Exits the config-ap-group context without saving changes.

## **show**

To display current AP group configuration settings, use the following command from within the config-ap-group context:

```
show
```

## **Defaults**

None

## **Example**

```
ruckus(config)# ap-group apgroup1
```

The AP group 'apgroup1' has been created. To save the AP group, type 'end' or 'exit'.

```
ruckus(config-apgrp)# show
```

```
APGROUP:
```

```
  ID:
```

```
  :
```

```
    Name= apgroup1
```

```
    Description=
```

```
    Radio 11bgn:
```

```
      Channelization= Auto
```

```
      Channel= Auto
```

```
    Enable auto channel selection which select from 1,6,11= Yes
```

```
    Tx. Power= Auto
```

```
    11N only Mode= Auto
```

```
WLAN Group= Default
Call Admission Control= OFF
Radio 11an:
Channelization= Auto
Channel= Auto
Tx. Power= Auto
11N only Mode= Auto
WLAN Group= Default
Call Admission Control= OFF
Members:
```

```
ruckus(config-apgrp) #
exit ???
```

## description

To set the AP group description, use the following command:

```
description <WORD>
```

## no description

To delete the AP group description, use the following command:

```
no description
```

## Configure Location Based Service Commands

Use the following commands to create and configure location services for an AP group. Use the `location-services` command to enter the `config-location-services` context from within the `config` context.

### location-services

To create and begin configuring location services for this AP group, use the following command:

```
location-services <WORD>
```

### Syntax Description

---

help	Set the IP addressing mode
------	----------------------------

---

history	IPv4, IPv6 or dual
abort	Exits the config-location-services context without saving changes.
end	Saves changes, and then exits the config-location-services context.
exit	Saves changes, and then exits the config-location-services context.
quit	Exits the config-location-services context without saving changes.
fqdn <WORD>	Sets the location server FQDN.
port <PORT-NUM>	Sets the location server port.
password <WORD>	Sets the location server preshared key.
show	Displays configured location services for all venues.

### Example

```
ruckus(config)# location-services locationservice1
```

The location venue 'locationservice1' has been created. To save it, type 'end' or 'exit'.

```
ruckus(config-location-services)# fqdn example1.ruckuswireless.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-location-services)# port 8883
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-location-services)# password password
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-location-services)# end
```

The location venue 'locationservice1' has been updated and saved. Your changes have been saved.

```
ruckus(config)#
```

### no location-services

To disable location-based service on this AP group, use the following command:

```
no location-services <WORD>
```

### **Example**

```
ruckus(config)# no location-service locationservice1
The location venue 'locationservice1' has been deleted.
ruckus(config)#
```

## **ipmode**

To set the IP addressing mode of the AP group, use the following command:

```
ipmode <WORD>
```

### **Syntax Description**

ipmode	Set the IP addressing mode
<WORD>	IPv4, IPv6 or dual

### **Example**

```
ruckus(config-apgrp)# ipmode dual
ruckus(config-apgrp)#
```

## **no ipmode-override**

To disable the override of IP mode, use the following command:

```
no ipmode-override
```

## **channelflyoff**

The ChannelFly override setting allows APs to disable ChannelFly if the AP's uptime is higher than the specified value (in minutes). To enable the ChannelFly override feature for the AP group, use the following command:

### **Defaults**

Disabled

30 minutes

**Example**

```
ruckus(config-apgrp) # channelflyoff 30
ruckus(config-apgrp) # show
APGROUP:
  ID:
  :
  Name= apgroup2
  Description=
  Channel Range:
    B/G/N= 1,2,3,4,5,6,7,8,9,10,11 (Disallowed= )
    A/N Indoor= 36,40,44,48,149,153,157,161 (Disallowed= )
    A/N Outdoor= 36,40,44,48,149,153,157,161 (Disallowed= )
  Radio 11bgn:
    Channelization= Auto
    Channel= Auto
    Tx. Power= Auto
    11N only Mode= Auto
    WLAN Group= Default
    Call Admission Control= OFF
    SpectraLink Compatibility= Disabled
  Radio 11an:
    Channelization= Auto
    Indoor Channel= Auto
    Outdoor Channel= Auto
    Tx. Power= Auto
    11N only Mode= Auto
    WLAN Group= Default
    Call Admission Control= OFF
    SpectraLink Compatibility= Disabled
  Network Setting:
    Protocol mode= Use Parent Setting
  Turn off channfly setting: enabled
    if AP's uptime is more than 30 minutes will turn off
  AP's ChannelFly
  Members:

ruckus(config-apgrp) #
```

## no channelflyoff

To disable the ChannelFly off feature for the AP group, use the following command:

```
no channelflyoff
```

## no channelflyoff-override

To disable the override of ChannelFly settings (use parent settings), use the following command:

```
no channelflyoff-override
```

## Example

```
ruckus(config-apgrp) # no channelflyoff-override
ruckus(config-apgrp) # show
APGROUP:
  ID:
  :
  Name= apgroup2
  Description=
  Channel Range:
    B/G/N= 1,2,3,4,5,6,7,8,9,10,11 (Disallowed= )
  A/N Indoor= 36,40,44,48,149,153,157,161 (Disallowed= )
  A/N Outdoor= 36,40,44,48,149,153,157,161 (Disallowed= )
  Radio 11bgn:
    Channelization= Auto
    Channel= Auto
    Tx. Power= Auto
    11N only Mode= Auto
    WLAN Group= Default
    Call Admission Control= OFF
    SpectraLink Compatibility= Disabled
  Radio 11an:
    Channelization= Auto
    Indoor Channel= Auto
    Outdoor Channel= Auto
    Tx. Power= Auto
    11N only Mode= Auto
    WLAN Group= Default
    Call Admission Control= OFF
```

```

SpectraLink Compatibility= Disabled
Network Setting:
  Protocol mode= Use Parent Setting
Turn off channfly setting: Use Parent Setting
Members:

```

```
ruckus(config-apgrp) #
```

## Radio 2.4/5 GHz Commands

Use the `radio 2.4` or `radio 5` commands to configure the 2.4/5 GHz radios on all APs within an AP group.

### radio

To configure radio settings for the 2.4 GHz or 5 GHz radios of an AP group, use the following command:

```
radio [2.4|5] <arguments>
```

### Syntax Description

<code>radio</code>	Configure AP group radio settings
<code>2.4</code>	Configure 2.4 GHz radio
<code>5</code>	Configure 5 GHz radio
<code>no</code>	Disables settings for the specified radios in the AP group
<code>channel</code>	Set radio channel (Auto or number)
<code>channelization</code>	Set radio channel width (Auto, 20MHz or 40MHz)
<code>auto-channel-selection</code> <code>[four-channel three-</code> <code>channel]</code>	Set auto channel selection to four-channel (1,5,9,13) or three-channel (1,6,11)
<code>tx-power</code>	Set radio transmit power (Auto, Full, 1/2, 1/4, 1/8, Min) or <NUMBER> (-1dB~-10dB)
<code>11n-only</code>	Set radio 11n-only mode to Auto or N-only
<code>wlan-group</code>	Set radio to the specified WLAN group

---

admission-control	Set the radio to use the specific call admission control airtime usage limit (%)
spectralink-compatibility	Enable SpectraLink Compatibility settings on the radio (sets DTIM=2, minrate=5.5Mbps and enable RTS-CTS protection mode)

---

### **Defaults**

Channel: Auto

Channelization: Auto

Auto-Channel Selection: Three-channel

TX Power: Auto

11n-only: Auto

WLAN group: Default

Admission Control: Off

SpecraLink Compatibility: Off

### **Example**

```
ruckus(config)# ap-group "System Default"
```

The AP group entry 'System Default' has been loaded. To save the AP group, type 'end' or 'exit'.

```
ruckus(config-apgrp)# radio 2.4 channel auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-apgrp)# radio 5 channelization auto
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-apgrp)# radio 5 11n-only N-only
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-apgrp)# radio 5 wlan-group Default
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-apgrp)# radio 2.4 tx-power Num 1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-apgrp)# show
```



```
APGROUP:
  ID:
  1:
  Name= System Default
  Description= System default group for Access Points
  Radio 11bgn:
  Channelization= Auto
  Channel= Auto
  Enable auto channel selection which select from 1,6,11= Yes
  Tx. Power= -1dB
  11N only Mode= Auto
  WLAN Group= Default
  Radio 11an:
  Channelization= Auto
  Channel= Auto
  Tx. Power= Auto
  11N only Mode= N-only
  WLAN Group= Default
  Members:
  MAC= 04:4f:aa:0c:b1:00
  MAC= 00:24:82:3f:14:60
  MAC= 74:91:1a:2b:ff:a0
  MAC= 00:1f:41:2a:2b:10

ruckus(config-apgrp)# end
The AP group 'System Default' has been updated.
Your changes have been saved.
ruckus(config)#
```

## radio 2.4 channel auto

Sets the 2.4GHz radio to use 'Auto' channel.

## radio 2.4 channel number <NUMBER>

Sets the 2.4GHz radio to use the specified channel.

## radio 2.4 channelization auto

Sets the 2.4GHz radio to use 'Auto' channelization.

**radio 2.4 channelization number <NUMBER>**

Sets the 2.4GHz radio to use the specified channelization.

**radio 2.4 auto-channel-selection four-channel**

Enables the auto channel selection which always select from 1,5,9,13.

**radio 2.4 auto-channel-selection three-channel**

Enables the auto channel selection which always select from 1,6,11.

**radio 2.4 tx-power Auto**

Sets the 2.4GHz radio to use 'Auto' Tx. power setting.

**radio 2.4 tx-power Full**

Sets the 2.4GHz radio to use the specified Tx. power setting.

**radio 2.4 tx-power 1/2**

Sets the 2.4GHz radio to use the specified Tx. power setting.

**radio 2.4 tx-power 1/4**

Sets the 2.4GHz radio to use the specified Tx. power setting.

**radio 2.4 tx-power 1/8**

Sets the 2.4GHz radio to use the specified Tx. power setting.

**radio 2.4 tx-power Min**

Sets the 2.4GHz radio to use the specified Tx. power setting.

**radio 2.4 tx-power Num**

Sets the 2.4GHz radio to use the specified Tx by number from 1-10 (-1dB ~ -10dB).

**radio 2.4 11n-only Auto**

Sets the 2.4GHz radio to use 'Auto' 11N only mode.

**radio 2.4 11n-only N-only**

Sets the 2.4GHz radio to use the specified 11N only mode.

**radio 2.4 wlan-group <WORD>**

Assigns the 2.4GHz radio to the specified WLAN group.

**radio 2.4 admission-control <VALUE>**

Sets the 2.4GHz radio to use the specific call admission control airtime usage limit(%).

**radio 2.4 spectralink-compatibility [enable | disable]**

Enables the SpectraLink Compatibility on 2.4GHz radio (will set DTIM=2, minrate=5.5Mbps and enable RTS-CTS protection mode).

**radio 2.4 channel-range <NUMBER-LIST>**

Sets the allowed list of channels used in 2.4GHz radio.

**radio 5 indoor channel auto**

Sets the 5GHz radio (indoor) to use 'Auto' channel.

**radio 5 indoor channel number <NUMBER>**

Sets the 5GHz radio (indoor) to use the specified channel.

**radio 5 indoor channel-range <NUMBER-LIST>**

Sets the allowed list of indoor channels used in 5GHz radio.

**radio 5 outdoor channel auto**

Sets the 5GHz radio (outdoor) to use 'Auto' channel.

**radio 5 outdoor channel number <NUMBER>**

Sets the 5GHz radio (outdoor) to use the specified channel.

**radio 5 outdoor channel-range <NUMBER-LIST>**

Sets the allowed list of outdoor channels used in 5GHz radio.

**radio 5 channel auto**

Sets the 5GHz radio to use 'Auto' channel.

**radio 5 channel number <NUMBER>**

Sets the 5GHz radio to use the specified channel.

**radio 5 channelization auto**

Sets the 5GHz radio to use 'Auto' channelization.

**radio 5 channelization number <NUMBER>**

Sets the 5GHz radio to use the specified channelization.

**radio 5 tx-power Auto**

Sets the 5GHz radio to use 'Auto' Tx. power setting.

**radio 5 tx-power Full**

Sets the 5GHz radio to use the specified Tx. power setting.

**radio 5 tx-power 1/2**

Sets the 5GHz radio to use the specified Tx. power setting.

**radio 5 tx-power 1/4**

Sets the 5GHz radio to use the specified Tx. power setting.

**radio 5 tx-power 1/8**

Sets the 5GHz radio to use the specified Tx. power setting.

**radio 5 tx-power Min**

Sets the 5GHz radio to use the specified Tx. power setting.

**radio 5 tx-power Num**

Sets the 5GHz radio to use the specified Tx by number from 1-10 (-1dB ~ -10dB).

**radio 5 11n-only Auto**

Sets the 5GHz radio to use 'Auto' 11N only mode.

**radio 5 11n-only N-only**

Sets the 5GHz radio to use the specified 11N only mode.

**radio 5 wlan-group <WORD>**

Assigns the 5GHz radio to the specified WLAN group.

**radio 5 admission-control <VALUE>**

Sets the 5GHz radio to use the specific call admission control airtime usage limit(%).

**radio 5 spectralink-compatibility [enable | disable]**

Enables the SpectraLink Compatibility on 5GHz radio (will set DTIM=2, minrate=5.5Mbps and enable RTS-CTS protection mode).

**no radio 2.4 channelization-override**

Disables the override of the 2.4GHz channelization settings.

**no radio 2.4 channel-range-override**

Disables the override of the 2.4GHz channel range settings.

**no radio 2.4 channel-override**

Disables the override of the 2.4GHz channel settings.

**no radio 2.4 tx-power-override**

Disables the override of the 2.4GHz Tx. power settings.

**no radio 2.4 11n-only-override**

Disables the override of the 2.4GHz 11N only mode settings.

**no radio 2.4 wlan-group-override**

Disables the override of the 2.4GHz WLAN group settings.

**no radio 2.4 admission-control**

Disables call admission control function on the 2.4GHz radio.

**no radio 2.4 admission-control-override**

Disables the override of the 2.4GHz call admission control settings.

**no radio 2.4 spectralink-compatibility-override**

Disables the override of the 2.4GHz SpectraLink Compatibility settings.

**no radio 5 indoor channel-range-override**

Disables the override of the 5GHz indoor channel range settings.

**no radio 5 indoor channel-override**

Disables the override of the 5GHz indoor channel settings.

**no radio 5 outdoor channel-range-override**

Disables the override of the 5GHz outdoor channel range settings.

**no radio 5 outdoor channel-override**

Disables the override of the 5GHz outdoor channel settings.

**no radio 5 channelization-override**

Disables the override of the 5GHz channelization settings.

**no radio 5 tx-power-override**

Disables the override of the 5GHz Tx. power settings.

**no radio 5 11n-only-override**

Disables the override of the 5GHz 11N only mode settings.

**no radio 5 wlan-group-override**

Disables the override of the 5GHz WLAN group settings.

**no radio 5 admission-control**

Disables call admission control function on the 5GHz radio.

**no radio 5 admission-control-override**

Disables the override of the 5GHz call admission control settings.

**no radio 5 spectralink-compatibility-override**

Disables the override of the 5GHz SpectraLink Compatibility settings.

## QoS Commands

Use the following commands to configure QoS settings for the AP group.

### **qos**

Contains commands that can be executed from within the context.

### **qos mld-query**

Contains commands that can be executed from within the context.

### **qos mld-query v1**

Enables the mld-query v1.

### **qos mld-query v2**

Enables the mld-query v2.

### **qos igmp-query**

Contains commands that can be executed from within the context.

### **qos igmp-query v2**

Enables the igmp-query v2.

### **qos igmp-query v3**

Enables the igmp-query v3.

### **no qos mld-query v1**

Disables the mld-query v1.

### **no qos mld-query v2**

Disables the mld-query v2.

### **no qos igmp-query v2**

Disables the igmp-query v2.

## no qos igmp-query v3

Disables the igmp-query v3.

## Model-Specific Commands

The following commands are used to configure model-specific settings for all APs of a certain model within an AP group.

### no model-setting

To discard the model settings for this specified model, use the following command:

```
no model-setting <WORD>
```

### model

To configure model-specific settings for all APs of a certain model within an AP group, use the following command:

```
model <WORD> <arguments>
```

### Syntax Description

model	Configure AP group model-specific settings
<WORD>	Enter the AP model name (e.g., zf2741, zf2741-ext, zf2942, zf7025, zf7055, zf7321, zf7321-u, zf7341, zf7343, zf7351,zf7352,zf7363,zf7372,zf7372-e, zf7441, zf7761cm, zf7762, zf7762-ac, zf7762-s, zf7762-s-ac, zf7762-t, zf7781-m, zf7781cm, zf7782, zf7782-e, zf7782-n, zf7782-s, zf7962, zf7982, sc8800-s, sc8800-s-ac, R300)
port-setting	Configures the port setting for the specified AP model. Enters config-apgrp-port context. See <a href="#">“Configure AP Group Model-Specific Port Settings”</a> for more information.
status-leds	Configures the status LEDs for the specified AP model (enable, disable).
usb-port	Configures the USB port settings for the AP model (enable, disable).



external-antenna	Configures external antenna settings. See “ <a href="#">Configure AP Group Model-Specific Antenna Settings</a> ”.
spectra-analysis	Configures spectrum analysis per radio (2.4Ghz / 5GHz, enable / disable).
radio-band	Sets the radio band for the AP group (APs with radio band selection only).
max-clients <NUMBER>	Sets the maximum clients for the AP.
usb-software <VID- PID-VERSION>	Selects the USB Software Vendor ID, Product ID and version for the AP.
poe-out	Configures the PoE Out ports for the specified AP model (enable, disable).
internal-heater	Configures the internal heater for the specified AP model (enable, disable).
cband-channels	Configures the C-band (5.8 GHz) channels for the specified AP model (enable, disable). (UK country code only)

### **Defaults**

Status LEDs: Enabled

PoE Out: Disabled

Internal Heater: Disabled

C-band channels: Disabled

USB Ports: Enabled

### **Example**

```
ruckus(config-apgrp) # model zf7343 status-leds enable
ruckus(config-apgrp) # end
The AP group 'System Default' has been updated.
Your changes have been saved.
ruckus(config) #
```

## Configure AP Group Model-Specific Antenna Settings

Use the `model <WORD> external-antenna` commands from within the `config-apgrp` context to configure model-specific external antenna settings for all APs of the specified model within the AP group. The following commands are available from within this context.

<code>external-antenna 2.4Ghz(11BG) enable</code>	Enables the external antenna setting for the 2.4GHz(11BG) radio.
<code>external-antenna 2.4Ghz(11BG) disable</code>	Disables the external antenna setting for the 2.4GHz(11BG) radio.
<code>external-antenna 2.4Ghz(11BG) gain</code>	Sets the external antenna gain for the 2.4GHz(11BG) radio.
<code>external-antenna 2.4Ghz(11BG) 2-antennas</code>	Selects the two external antennas for the 2.4GHz(11BG) radio.
<code>external-antenna 2.4Ghz(11BG) 3-antennas</code>	Selects the three external antennas for the 2.4GHz(11BG) radio.
<code>external-antenna 2.4Ghz(11NG) enable</code>	Enables the external antenna setting for the 2.4GHz(11NG) radio.
<code>external-antenna 2.4Ghz(11NG) disable</code>	Disables the external antenna setting for the 2.4GHz(11NG) radio.
<code>external-antenna 2.4Ghz(11NG) gain</code>	Sets the external antenna gain for the 2.4GHz(11NG) radio.
<code>external-antenna 2.4Ghz(11NG) 2-antennas</code>	Selects the two external antennas for the 2.4GHz(11NG) radio.
<code>external-antenna 2.4Ghz(11NG) 3-antennas</code>	Selects the three external antennas for the 2.4GHz(11NG) radio.
<code>external-antenna 5Ghz(11NA) enable</code>	Enables the external antenna setting for the 5GHz(11NA) radio.
<code>external-antenna 5Ghz(11NA) disable</code>	Disables the external antenna setting for the 5GHz(11NA) radio.

external-antenna 5Ghz(11NA) gain	Sets the external antenna gain for the 5GHz(11NA) radio.
external-antenna 5Ghz(11NA) 2- antennas	Selects the two external antennas for the 2.4GHz(11NA) radio.
external-antenna 5Ghz(11NA) 3- antennas	Selects the three external antennas for the 2.4GHz(11NA) radio.
external-antenna 5Ghz(11A) enable	Enables the external antenna setting for the 5GHz(11A) radio.
external-antenna 5Ghz(11A) disable	Disables the external antenna setting for the 5GHz(11A) radio.
external-antenna 5Ghz(11A) gain	Sets the external antenna gain for the 5GHz(11A) radio.
external-antenna 5Ghz(11A) 2-antennas	Selects the two external antennas for the 2.4GHz(11A) radio.
external-antenna 5Ghz(11A) 3-antennas	Selects the three external antennas for the 2.4GHz(11A) radio.

## Configure AP Group Model-Specific Port Settings

Use the `model <WORD> port-setting` command (from the `config-apgrp` context) to enter the `config-apgrp-port` context and configure model-specific port settings for all APs of the specified model within the AP group. The following commands are available from within this context.

port-setting	Enters the port-setting context.
no port-setting	Disables the override of the global AP mode configuration.
help	Shows available commands.
history	Shows a list of previously run commands.
abort	Exits the config-apgrp-port context without saving changes.
end	Saves changes, and then exits the config-apgrp-port context.

exit	Saves changes, and then exits the config-apgrp-port context.
quit	Exits the config-apgrp-port context without saving changes.
show	Displays config-apgrp-port context.
lan <NUMBER>	Enables the AP Ethernet port.
lan <NUMBER> uplink <WORD>	Sets the AP port to use the specified type (trunk, access or general).
lan <NUMBER> untag <NUMBER>	Sets the AP port to use the specified VLAN ID(1-4094).
lan <NUMBER> member <NUMBER>	Sets the AP port to use the specified members(1-4094).
lan <NUMBER> opt82 enabled	Enables the AP port DHCP option 82 settings.
lan <NUMBER> opt82 disabled	Disables the AP port DHCP option 82 settings.
lan <NUMBER> tunnel disabled	Disables the AP port tunnel settings.
lan <NUMBER> tunnel enabled	Enables the AP port tunnel settings.
lan <NUMBER> dot1x disabled	Disables the AP port 802.1X settings.
lan <NUMBER> dot1x supplicant	Sets the AP port to 802.1X supplicant.
lan <NUMBER> dot1x auth-port-based	Sets the AP port to port-based 802.1X.
lan <NUMBER> dot1x auth-mac-based	Sets the AP port to mac-based 802.1X.
lan <NUMBER> guest-vlan <WORD>	Sets the AP port to use the specified guest VLAN ID(1-4094).
lan <NUMBER> dvlan enabled	Enables the AP port dynamic VLAN settings.
lan <NUMBER> dvlan disabled	Disables the AP port dynamic VLAN settings.

lan <NUMBER> qos mld-snooping	Enables the AP port MLD Snooping setting.
lan <NUMBER> qos igmp-snooping	Enables the AP port IGMP Snooping setting.
dot1x supplicant mac	Sets the username and password to use AP MAC address for AP 802.1X supplicant.
dot1x supplicant user- name <WORD>	Sets the username for AP 802.1X supplicant.
dot1x supplicant user- name <WORD> password <WORD>	Sets the password for AP 802.1X supplicant.
dot1x authsvr <WORD>	Sets the authentication server for AP 802.1X.
dot1x acctsvr <WORD>	Sets the accounting server for AP 802.1X.
dot1x mac-auth- bypass	Enables MAC authentication bypass (Use device MAC address as username and password).
no lan <NUMBER>	Disables the AP Ethernet port.
no dot1x authsvr	Disables the auth server settings.
no lan <NUMBER> qos mld-snooping	Disables the AP port MLD Snooping setting.
no lan <NUMBER> qos igmp-snooping	Disables the AP port IGMP snooping setting.
no dot1x authsvr	Disables the authentication server settings.
no dot1x acctsvr	Disables the accounting server settings.
no dot1x mac-auth- bypass	Disables the MAC authentication bypass.

### **Example**

```
ruckus(config-apgrp) # model zf7372 port-setting
ruckus(config-apgrp-port) # show
PORTS:
    LAN ID:
        1:
```

```

Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
Guest VLAN=
Enable Dynamic VLAN= Disabled
802.1X= disabled
DHCP opt82= Disabled
Tunnel= Disabled
MLD Snooping= Disabled
IGMP Snooping= Enabled
2:
Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
Guest VLAN=
Enable Dynamic VLAN= Disabled
802.1X= disabled
DHCP opt82= Disabled
Tunnel= Disabled
MLD Snooping= Disabled
IGMP Snooping= Enabled
ruckus(config-apgrp-port)#

```

## AP Group Membership

Use the following commands to configure AP group membership (move APs into or out of the current AP group, from within the config-apgrp context).

### member

Adds or moves the AP to the specified AP group.

```
member [add|move] mac <WORD> [system-default|name <WORD>]
```

### member add mac

To add the AP to the specified AP group, use the following command:

```
member add mac <WORD>
```

**Example**

```
ruckus(config-apgrp) # member add mac c4:10:8a:1f:d1:f0
ruckus(config-apgrp) # show
APGROUP:
  ID:
  :
  Name= apgroup2
  Description=
  Channel Range:
    B/G/N= 1,2,3,4,5,6,7,8,9,10,11 (Disallowed= )
    A/N Indoor= 36,40,44,48,149,153,157,161 (Disallowed= )
    A/N Outdoor= 36,40,44,48,149,153,157,161 (Disallowed= )
  Radio 11bgn:
    Channelization= Auto
    Channel= Auto
    Tx. Power= Auto
    11N only Mode= Auto
    WLAN Group= Default
    Call Admission Control= OFF
    SpectraLink Compatibility= Disabled
  Radio 11an:
    Channelization= Auto
    Indoor Channel= Auto
    Outdoor Channel= Auto
    Tx. Power= Auto
    11N only Mode= Auto
    WLAN Group= Default
    Call Admission Control= OFF
    SpectraLink Compatibility= Disabled
  Network Setting:
    Protocol mode= Use Parent Setting
  Turn off channfly setting: disabled
  if AP's uptime is more than 30 minutes will turn off
  AP's ChannelFly
  Members:
    MAC= c4:10:8a:1f:d1:f0

ruckus(config-apgrp) #
```

## member mac move-to system-default

To move the AP from the current AP group to the System Default AP group, use the following command:

```
member mac <WORD> move-to system-default
```

### **Example**

```
ruckus(config-apgrp) # member mac c4:10:8a:1f:d1:f0 move-to system-  
default  
ruckus(config-apgrp) #
```

## member mac move-to name

To move the AP from the current AP group to the specified AP group, use the following command:

```
member mac <WORD> move-to name <WORD>
```

### **Example**

```
ruckus(config-apgrp) # member mac c4:10:8a:1f:d1:f0 move-to name  
apgroup2  
ruckus(config-apgrp) #
```

## Model-Specific Port Settings

This section describes the commands that you can use to configure port settings for all APs of a specific model within an AP group. The following commands can be executed from within the `config-apgrp-port` context. To show a list of commands available from within the context, type `help` or `?`.

### **model port-setting**

To configure the port settings for all APs of a specific model within an AP group, and enter the `config-apgrp-port` context, use the following command:

```
model <WORD> port-setting
```



## Syntax Description

model	Configure AP group model-specific settings
<WORD>	Enter the AP model name (e.g., zf2942, zf2741, zf7025, zf7341, zf7343, zf7363, zf7761cm, zf7762, zf7762-s, zf7762-t, zf7762-ac, zf7762-s-ac, zf7762-t-ac, zf7942, zf7962).
port-setting	Configures the port setting for the specified AP model. Enters config-apgrp-port context.

## Example

```
ruckus(config)# ap-group "System Default"
The AP group entry 'System Default' has been loaded. To save the
AP group, type 'end' or 'exit'.
ruckus(config-apgrp)# model zf7025 port-setting
ruckus(config-apgrp-port)#
```

## abort

To exit the config-apgrp-port context without saving changes, use the following command:

```
abort
```

## Syntax Description

abort	Exit the context without saving changes
-------	---

## Defaults

None.

## Example

```
ruckus(config-apgrp-port)# abort
ruckus(config-apgrp)#
```

## end

To save changes, and then exit the `config-apgrp-port` context, use the following command:

```
end
```

### **Syntax Description**

---

<code>end</code>	Save changes, and then exit the context
------------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-apgrp-port) # end  
ruckus(config-apgrp) #
```

## exit

To save changes, and then exit the `config-apgrp-port` context, use the following command:

```
exit
```

### **Syntax Description**

---

<code>exit</code>	Save changes, and then exit the context
-------------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-apgrp-port) # exit  
ruckus(config-apgrp) #
```

## quit

To exit the `config-apgrp-port` context without saving changes, use the following command:

```
quit
```

### **Syntax Description**

---

<code>quit</code>	Exit the context without saving changes
-------------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-apgrp-port)# quit  
ruckus(config-apgrp)#
```

## show

To show a device's port state, use the following command:

```
show
```

### **Syntax Description**

---

<code>show</code>	Display the device's port state
-------------------	---------------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(config-apgrp)# model zf7962 port-setting  
ruckus(config-apgrp-port)# show  
PORTS:  
LAN ID:  
1:
```

```

Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
802.1X= disabled
DHCP opt82= Disabled
LAN ID:
2:
Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
802.1X= disabled
DHCP opt82= Disabled
ruckus(config-apgrp-port) #

```

## no lan

To disable a LAN port on APs in an AP group, use the following command:

```
no lan <NUMBER>
```

## Syntax Description

no lan	Disable a specific port
<NUMBER>	Disable this port

## Defaults

Enabled.

## Example

```

ruckus(config-apgrp-port) # no lan 2
ruckus(config-apgrp-port) #

```

## lan

To enable a LAN port on APs in an AP group, use the following command:

```
lan <NUMBER>
```

### **Syntax Description**

lan	Enable a specific port
<NUMBER>	Enable this port

### **Defaults**

Enabled.

### **Example**

```
ruckus(config-apgrp-port) # lan 2
ruckus(config-apgrp-port) #
```

## **lan uplink**

To set port type, use the following command:

```
lan <NUMBER> uplink <WORD>
```

### **Syntax Description**

lan	Configure a specific port
<NUMBER>	Configure this port
uplink	Set the port type
<WORD>	Port type (Trunk port, Access port, General port)

### **Defaults**

All AP ports other than ZF 7025: Trunk

ZF 7025 port 5: Trunk

ZF 7025 LAN 1-LAN 4: Access

### **Example**

```
ruckus(config-apgrp) # model zf7962 port-setting
```

```
ruckus(config-apgrp-port) # lan 2 uplink access
ruckus(config-apgrp-port) # show
PORTS:
LAN ID:
1:
Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
802.1X= disabled
DHCP opt82= Disabled
LAN ID:
2:
Enable LAN = Yes
LAN Type= access
Untag ID= 1
Members= 1
802.1X= disabled
DHCP opt82= Disabled
ruckus(config-apgrp-port) #
```

## lan untag

To configure untag VLAN settings for a model-specific port, use the following command:

```
lan <NUMBER> untag <NUMBER>
```

### ***Syntax Description***

lan untag	Configure port untag VLAN
<NUMBER>	Configure this port
<NUMBER>	Set untag VLAN to this number

### ***Defaults***

1

## Example

```
ruckus(config-apgrp-port)# lan 2 untag 20
ruckus(config-apgrp-port)#
```

## lan member

To set the LAN port VLAN membership (only General ports have configurable membership; Trunk ports are members of all VLANs, and Access port membership must be the same as the Untag VLAN), use the following command:

```
lan <NUMBER> member <NUMBER>
```

## Syntax Description

lan member	Set the LAN port VLAN membership
<NUMBER>	Specify the LAN port to configure
<NUMBER>	Set the VLAN membership (1~4094, range separated by hyphen, multiple VLANs separated by commas)

## Defaults

1

## Example

```
ruckus(config-apgrp-port)# lan 2 uplink general
ruckus(config-apgrp-port)# lan 2 member 1-10,100,200
ruckus(config-apgrp-port)# show
PORTS:
LAN ID:
1:
Enable LAN = Yes
LAN Type= trunk
Untag ID= 1
Members= 1-4094
802.1X= disabled
DHCP opt82= Disabled
LAN ID:
```

```

2:
Enable LAN = Yes
LAN Type= general
Untag ID= 20
Members= 1-10,100,200
802.1X= disabled
DHCP opt82= Disabled
ruckus(config-apgrp-port) #

```

## lan opt82

To enable or disable DHCP option 82 for a LAN port, use the following command:

```
lan <NUMBER> opt82 [enable|disable]
```

### *Syntax Description*

lan opt82	Enable or disable DHCP option 82
enable	Enable option 82
disable	Disable option 82

### *Defaults*

Disabled

### *Example*

```

ruckus(config-apgrp-port) # lan 2 opt82 enable
ruckus(config-apgrp-port) #

```

## dot1x

To enable 802.1X on ports of all APs of a specific model in an AP group, use the following command:

```

model <WORD> dot1x
lan <NUMBER> dot1x [disable|supplicant|auth-port-
based|auth-mac-based|guest-vlan<NUMBER>|dvlan]

```



## Syntax Description

lan dot1x	Configure 802.1X settings for this port
<NUMBER>	LAN port number to configure
disable	Disable 802.1X
supplicant	Configure this LAN port as an 802.1X supplicant
auth-port-based	Configure this LAN port as an 802.1X authenticator (port-based)
auth-mac-based	Configure this LAN port as an 802.1X authenticator (MAC-based)

## Defaults

Disabled

## Example

```
ruckus(config-apgrp) # model zf7025 port-setting
ruckus(config-apgrp-port) # lan 1 dot1x supplicant
ruckus(config-apgrp-port) # show
PORTS:
LAN ID:
1:
Enable LAN = Yes
LAN Type= access
Untag ID= 1
Members= 1
802.1X= supp
DHCP opt82= Disabled
```

## dot1x authsvr

To configure 802.1X authentication server, use the following command:

```
dot1x authsvr <WORD>
```

## Syntax Description

dot1x authsvr	Configure 802.1X authentication server
<WORD>	Name of AAA server

## Defaults

None

## Example

```
ruckus(config-apgrp-port) # dot1x authsvr radius
ruckus(config-apgrp-port) #
```

## dot1x acctsvr

To configure 802.1X accounting server, use the following command:

```
dot1x acctsvr <WORD>
```

## Syntax Description

dot1x acctsvr	Configure 802.1X accounting server
<WORD>	Name of AAA server

## Defaults

None

## Example

```
ruckus(config-apgrp-port) # dot1x acctsvr radius-acct
ruckus(config-apgrp-port) #
```

## dot1x mac-auth-bypass

To configure 802.1X MAC authentication bypass, use the following command:

```
dot1x mac-auth-bypass
```

## Syntax Description

---

dot1x mac-auth-bypass	Enable 802.1X MAC authentication bypass
-----------------------	---

---

## Defaults

Disabled

## Example

```
ruckus(config-apgrp-port) # dot1x mac-auth-bypass
ruckus(config-apgrp-port) #
```

## dot1x supplicant username

To configure 802.1X supplicant user name, use the following command:

```
dot1x supplicant username <WORD>
```

## Syntax Description

---

dot1x supplicant username	Configure 802.1X supplicant user name
<WORD>	Set the 802.1X supplicant user name

---

## Defaults

None

## Example

```
ruckus(config-apgrp-port) # dot1x supplicant username johndoe
ruckus(config-apgrp-port) #
```

## dot1x supplicant password

To configure 802.1X supplicant password, use the following command:

```
dot1x supplicant password <WORD>
```

## Syntax Description

dot1x supplicant password	Configure 802.1X supplicant password
<WORD>	Set the 802.1X supplicant password

## Defaults

None

## Example

```
ruckus(config-apgrp-port) # dot1x supplicant password test123
ruckus(config-apgrp-port) #
```

## dot1x supplicant mac

To set the 802.1X supplicant user name and password as the AP's MAC address, use the following command:

```
dot1x supplicant mac
```

## Syntax Description

dot1x supplicant mac	Set the supplicant user name and password as the AP's MAC address
----------------------	---

## Defaults

None

## Example

```
ruckus(config-apgrp-port) # dot1x supplicant mac
ruckus(config-apgrp-port) #
```

## no dot1x

To disable 802.1X settings for an AP model, use the following command:

```
no dot1x [authsvr] [acctsvr] [mac-auth-bypass]
```

## Syntax Description

---

no dot1x	Disable dot1x settings for the AP
authsvr	Disable authentication server
acctsvr	Disable accounting server
mac-auth-bypass	Disable MAC authentication bypass

---

## Defaults

None

## Example

```
ruckus(config-apgrp-port)# no dot1x authsvr
ruckus(config-apgrp-port)#
```

## lan guest-vlan

To set the AP port to use the specified guest VLAN ID(1-4094), use the following command:

```
lan <NUMBER> guest-vlan <WORD>
```

## lan dvlan

To enable/disable dynamic VLAN for the AP port, use the following command:

```
lan <NUMBER> dvlan [enabled | disabled]
```

## lan qos

To set the AP port QoS settings, use the following command:

```
lan <NUMBER> qos
```

## lan qos mld-snooping

To enable MLD snooping for the port, use the following command:

```
lan <NUMBER> qos mld-snooping
```

## lan qos igmp-snooping

To enable IGMP snooping for the port, use the following command:

```
lan <NUMBER> qos igmp-snooping
```

### **no lan qos**

To disable QoS settings for the port, use the following command:

```
no lan <NUMBER> qos
```

### **no lan qos mld-snooping**

To disable MLD snooping on the port, use the following command:

```
no lan <NUMBER> qos mld-snooping
```

### **no lan qos igmp-snooping**

To disable IGMP snooping on the port, use the following command:

```
no lan <NUMBER> qos igmp-snooping
```

### **no dot1x**

To disable 802.1x settings for the port, use the following command:

```
no dot1x
```

### **no dot1x authsvr**

To disable the authentication server settings, use the following command

```
no dot1x authsvr
```

### **no dot1x acctsvr**

To disable the accounting server settings, use the following command:

```
no dot1x acctsvr
```

### **no dot1x mac-auth-bypass**

To disable MAC authentication bypass, use the following command:

```
no dot1x mac-auth-bypass
```

## **LLDP Commands**

To enable, disable or configure the Link Layer Discovery Protocol (LLDP) commands for the AP group, use the following commands from within the config-apgrp context.

## lldp

To enable, disable or configure the AP group's Link Layer Discover Protocol settings, use the following commands.

### Syntax Description

lldp	Configure LLDP settings.
enable	Enable LLDP with current settings.
disable	Disable LLDP with current settings.
interval <NUMBER>	Set packet transmit interval in second(s).
holdtime <NUMBER>	Set amount of time receiving device should retain the information.
ifname eth <NUMBER>	Enter the AP port number.
mgmt enable	Enable LLDP management IP address of the AP.
mgmt disable	Disable LLDP management IP address of the AP.

### Example

```
ruckus(config-apgrp) # lldp enable
ruckus(config-apgrp) #
```

## Configure Certificate Commands

Use the `config-certificate` commands to restore the default ZoneDirector certificate or to regenerate the private key. To run these commands, you must first enter the `config-certificate` context.

### quit

To exit the `config-certificate` context without saving changes, use the `quit` command.

```
quit
```

## Syntax Description

---

<code>quit</code>	Exit the certificate settings without saving changes
-------------------	--

---

### Defaults

None.

### Example

```
ruckus(config-certificate)# quit
```

No changes have been saved.

### restore

To restore the default ZoneDirector certificate and private key, use the following command.

```
restore
```

## Syntax Description

---

<code>restore</code>	Restore the default ZoneDirectory certificate and private key. The restore process will be completed after ZoneDirector is rebooted.
----------------------	--

---

### Defaults

None.

### Example

```
ruckus(config-certificate)# restore
```

ZoneDirector will restart now to apply the changes in the certificate settings. If you want to configure other settings, log in again after ZoneDirector has completed restarting.

### re-generate-private-key

To regenerate the ZoneDirector private key, use the following command:

```
re-generate-private-key {1024|2048}
```



## Syntax Description

---

re-generate-private-key	Regenerate the ZoneDirector private key
{1024 2048}	Specify the length of the private key as either 1024 or 2048.

---

### Defaults

None.

### Example

```
ruckus(config-certificate)# re-generate-private-key 1024  
ZoneDirector will restart now to apply the changes in the certificate  
settings. If you want to configure other settings, log in again  
after ZoneDirector has completed restarting.  
The operation doesn't execute successfully. Please try again.
```

## Configure Hotspot Redirect Settings

To configure Hotspot redirect settings, use the following command:

### hotspot\_redirect\_https

To enable Hotspot redirect, use the following command:

```
hotspot_redirect_https
```

### Defaults

None.

### Example

```
ruckus(config)# hotspot_redirect_https  
/bin/hotspot_redirect_https enable  
ruckus(config)#
```

### no hotspot\_redirect\_https

To disable Hotspot redirect, use the following command:

```
no hotspot_redirect_https
```

### **Defaults**

None.

### **Example**

```
ruckus(config)# no hotspot_redirect_https  
/bin/hotspot_redirect_https disable  
ruckus(config)#
```

### **no blocked-client**

To remove a blocked client from the blocked clients list, use the following command:

```
no blocked-client <MAC>
```

### **Defaults**

None.

### **Example**

```
ruckus(config)# no blocked-client dc:2b:61:13:f7:72  
The L2 ACL 'dc:2b:61:13:f7:72' has been deleted.  
ruckus(config)#
```

## **ConfigureLayer2AccessControlCommands**

Use the `layer2 access control` commands to configure the Layer 2 Access Control List settings. To run these commands, you must first enter the `config-l2acl` context.

### **acl**

To create a new L2 ACL entry or update an existing entry, use the following command:

```
acl <WORD>
```

## ***Syntax Description***

---

acl	Create a new ACL
<WORD>	Assign this name to the new ACL

---

## ***Defaults***

None.

## ***Example***

```
ruckus(config)# l2acl l2acl1  
The L2 ACL entry 'l2acl1' has been created.  
ruckus(config-l2acl)#
```

## **no acl**

To delete an L2 ACL, use the following command:

```
no acl <WORD>
```

## ***Syntax Description***

---

no acl	Delete an existing ACL
<WORD>	Delete this ACL

---

## ***Defaults***

None.

## ***Example***

```
ruckus(config)# no l2acl l2acl1  
The L2 ACL 'l2acl1' has been deleted.  
ruckus(config)#
```

## abort

To exit the `config-l2acl` context without saving changes, use the following command:

```
abort
```

### **Syntax Description**

---

abort	Exit the <code>config-l2acl</code> context without saving changes
-------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-l2acl)# abort  
No changes have been saved.  
ruckus(config)#
```

## end

To save changes, and then exit the `config-l2acl` context, use the following command:

```
end
```

### **Syntax Description**

---

end	Save changes and exit the <code>config-l2acl</code> context
-----	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-l2acl)# end  
The L2 ACL entry has saved successfully.  
Your changes have been saved.  
ruckus(config)#
```

## exit

To save changes, and then exit the `config-12acl` context, use the following command:

```
exit
```

### **Syntax Description**

---

exit	Save changes and exit the <code>config-12acl</code> context
------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-12acl)# exit  
The L2 ACL entry has saved successfully.  
Your changes have been saved.  
ruckus(config)#
```

## quit

To exit the `config-12acl` context without saving changes, use the following command:

```
quit
```

### **Syntax Description**

---

quit	Exit the <code>config-12acl</code> context without saving changes
------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-12acl)# quit  
No changes have been saved.
```

```
ruckus(config)#
```

## **show**

To displays the L2 ACL settings, use the `show` command. You must run this command from within the `config-l2acl` context.

```
show
```

### ***Syntax Description***

---

<code>show</code>	Display the Layer 2 access control list settings
-------------------	--

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-l2acl)# show
L2/MAC ACL:
  ID:
  :
  Name= l2acl1
  Description=
  Restriction= Deny only the stations listed below
  Stations:
    MAC Address= 00:11:22:33:44:55

ruckus(config-l2acl)#
```

## **name**

To rename an L2 ACL entry, use the following command:

```
name <WORD>
```

### ***Syntax Description***

---

<code>name</code>	Sets the L2 ACL entry name.
-------------------	-----------------------------

---

---

<WORD> Rename the ACL to this name.

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# l2acl l2acl1  
The L2 ACL entry 'l2acl1' has been created.  
ruckus(config-l2acl)# name L2-ACL-1  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-l2acl)#
```

### **description**

To set the description of an L2 ACL entry, use the following command (multiple word text must be enclosed in quotation marks):

```
description <WORD>
```

### **Syntax Description**

---

description <WORD> Set the L2 ACL description.

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# l2acl l2acl1  
The L2 ACL entry 'l2acl1' has been created.  
ruckus(config-l2acl)# description "L2 ACL 1"  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-l2acl)#
```

## add-mac

To add a MAC address to the L2 ACL, use the following command:

```
add-mac <MAC>
```

### Syntax Description

---

add mac	Add a MAC address to the ACL
<MAC>	Add this MAC address

---

### Defaults

None.

### Example

```
ruckus(config-l2acl)# add-mac 00:11:22:33:44:55  
The station '00:11:22:33:44:55' has been added to the ACL.  
ruckus(config-l2acl)#
```

## mode allow

To set the ACL mode to 'allow', use the following command:

```
mode allow
```

### Syntax Description

---

mode allow	Set the ACL mode to allow
------------	---------------------------

---

### Defaults

None.

### Example

```
ruckus(config-l2acl)# mode allow  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-l2acl)#
```



## mode deny

To set the ACL mode to 'deny', use the following command:

```
mode deny
```

### Syntax Description

---

mode deny	Set the ACL mode to deny
-----------	--------------------------

---

### Defaults

None.

### Example

```
ruckus(config-l2acl)# mode deny
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-l2acl)#
```

## del-mac

To delete a MAC address from an L2 ACL, use the following command:

```
del-mac <MAC>
```

### Syntax Description

---

del-mac	Delete a MAC address from the ACL
<MAC>	Delete this <MAC>

---

### Defaults

None.

### Example

```
ruckus(config-l2-acl)# del-mac 00:01:02:34:44:55
```

The station '00:01:02:34:44:55' has been removed from the ACL.

```
ruckus(config-l2-acl)# del-mac 00:01:02:34:44:55
```

The station '00:01:02:34:44:55' could not be found. Please check the spelling, and then try again.

## ConfigureLayer3AccessControlCommands

Use the `l3acl` commands to configure the Layer 3 Access Control List settings. To run these commands, you must first enter the `config-l3acl` or `config-l3acl-ipv6` context.

### **l3acl**

To enter the `config-l3acl` context, run this command:

```
l3acl <WORD>
```

### **Syntax Description**

---

<code>l3acl</code>	Create or configure a Layer 3 Access Control List
<code>&lt;WORD&gt;</code>	Name of the L3 ACL

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# l3acl "ACL 1"
```

The L3/L4/IP ACL entry 'ACL 1' has been created.

```
ruckus(config-l3acl)#
```

### **l3acl-ipv6**

To enter the `config-l3acl-ipv6` context, run this command:

```
l3acl-ipv6 <WORD>
```

### **Syntax Description**

---

<code>l3acl-ipv6</code>	Create or configure a Layer 3 Access Control List
<code>&lt;WORD&gt;</code>	Name of the L3 ACL

---

## Defaults

None.

## Example

```
ruckus(config)# l3acl-ipv6 "ACL 2"  
The L3/L4/IPv6 ACL entry 'ACL 2' has been created.  
ruckus(config-l3acl-ipv6)#
```

## no l3acl

To delete an L3/L4 ACL entry, use the following command:

```
no l3acl <WORD>
```

## Syntax Description

no l3acl	Delete a Layer 3 ACL
<WORD>	Name of the L3 ACL

## Defaults

None.

## Example

```
ruckus(config)# no l3acl "ACL test"  
The L3/L4/IP ACL 'ACL test' has been deleted.  
ruckus(config)#
```

## abort

To exit the config-l3acl context without saving changes, use the following command:

```
abort
```

## ***Syntax Description***

---

abort	Exit the context without saving changes
-------	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-l3acl)# abort  
No changes have been saved.  
ruckus(config)#
```

## **end**

To save changes, and then exit the config-l3acl context, use the following command:

```
end
```

## ***Syntax Description***

---

end	Save changes and exit the context
-----	-----------------------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-l3acl)# end  
The L3/L4/IP ACL entry has saved successfully.  
Your changes have been saved.  
ruckus(config)#
```

## **exit**

To save changes, and then exit the config-l3acl context, use the following command:

```
exit
```

## **Syntax Description**

---

<code>exit</code>	Save changes and exit the context
-------------------	-----------------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus# config-l3acl
ruckus(config-l3acl)# exit
Your changes have been saved.
```

### **quit**

To exit the `config-l3acl` context without saving changes, use the following command:

```
quit
```

## **Syntax Description**

---

<code>quit</code>	Exit the context without saving changes
-------------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-l3acl)# quit
No changes have been saved.
ruckus(config)#
```

### **show**

To display the L3ACL settings, use the `show` command. You must run this command from within the `config-l3acl` context.

```
show
```

## Syntax Description

---

show	Display the Layer 3 access control list settings
------	--

---

### Defaults

None.

### Example

```
ruckus(config-l3acl)# show
L3/L4/IP ACL:
ID:
3:
Name= test_newname
Description= justfortestCLI
Default Action if no rule is matched= Deny all by default
Rules:
Order= 1
Description=
Type= Allow
Destination Address= Any
Destination Port= 53
Protocol= Any
Order= 2
Description=
Type= Allow
Destination Address= Any
Destination Port= 67
Protocol= Any
```

### name

To set the name of anL3/L4/IP ACL entry, use the following command:

```
name <WORD>
```

## Syntax Description

---

name	Set the name of anL3/L4/IP ACL entry
------	--------------------------------------

---

---

<WORD>	Name of the L3/L4/IP ACL entry
--------	--------------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(config-l3acl)# name test_newname
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **description**

To set the description of an L3/L4/IP ACL entry, use the following command (multiple word text must be enclosed in quotes):

```
description <WORD>
```

### **Syntax Description**

---

description	Set the L3/L4/IP ACL entry description
<WORD>	Set to this description

---

### **Defaults**

None.

### **Example**

```
ruckus(config-l3acl)# description justfortestCLI
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **mode allow**

To set the ACL mode to 'allow', use the following command:

```
mode allow
```

## ***Syntax Description***

---

mode	Set the ACL mode
allow	Set the mode to 'allow'

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-l3acl)# mode allow
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **mode deny**

To set the ACL mode to 'deny', use the following command:

```
mode deny
```

## ***Syntax Description***

---

mode	Set the ACL mode
deny	Set the mode to 'deny'

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-l3acl)# mode deny
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **no rule-order**

To delete a rule from the L3/L4/IP ACL, use the following command:

```
no rule-order <NUMBER>
```



## ***Syntax Description***

---

no rule-order	Delete a rule from the L3/L4/IP ACL
<NUMBER>	Delete this rule ID

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-l3acl)# no rule-order 3  
The rule '3' has been removed from the ACL.
```

### **rule-order**

To create or modify a rule in the L3/L4/IP ACL, use the following command:

```
rule-order <NUMBER>
```

## ***Syntax Description***

---

rule-order	Create a new rule or modify an existing one
<NUMBER>	Create or modify this rule ID

---

### ***Defaults***

None.

### ***Example***

For example, to set the current rule as the third ACL rule to apply, use the following command:

```
ruckus(config-l3acl)# rule-order 3  
ruckus(config-l3acl-rule)#
```

## Layer 3 Access Control Rule Commands

Use the `l3acl-rule` commands to configure the Layer 3/Layer 4/IP Access Control List rules. To run these commands, you must first enter the `config-l3acl-rule` context. To enter the `config-l3acl-rule` context, run this command:

```
rule-order <NUMBER>
```

### end

To save changes, and then exit the `config-l3acl-rule` context, use the following command:

```
end
```

### exit

To save changes, and then exit the `config-l3acl-rule` context, use the following command:

```
exit
```

### order

To set the L3/L4/IP ACL rule order, use the following command:

```
order <NUMBER>
```

### Example

```
ruckus(config-l3acl-rule)# order 1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-l3acl-rule)#
```

### description

To set the description of an L3/L4/IP ACL rule, use the following command (multiple word text must be enclosed in quotes):

```
description <WORD>
```

### Syntax Description

---

description	Set the L3/L4/IP ACL rule description
<WORD>	Set to this description

---

## Defaults

None.

## Example

```
ruckus(config-l3acl-rule)# description thirdl3rule
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## type allow

To set the ACL rule type to 'allow', use the following command:

```
type allow
```

## Syntax Description

---

type	Set the ACL rule type
allow	Set the rule type to 'allow'

---

## Defaults

None.

## Example

```
ruckus(config-l3acl-rule)# type allow
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## type deny

To set the ACL rule type to 'deny', use the following command:

```
type deny
```

## Syntax Description

---

type	Set the ACL rule type
deny	Set the rule type to 'deny'

---

## Defaults

None.

## Example

```
ruckus(config-l3acl-rule)# type deny
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## destination address

To set the destination address of the rule, use the following command:

```
destination address <IP-ADDR/WORD>
```

## Syntax Description

destination address	Set the destination address of the rule
IP-ADDR/WORD	Set the destination to this IP address

## Defaults

None.

## Example

```
ruckus(config-l3acl-rule)# destination address 192.168.1.22
```

The destination IP address is invalid. Please enter 'Any' or check the IP address(for example:192.168.0.1/24), and then please try again.

```
ruckus(config-l3acl-rule)# destination address 192.168.1.22/24
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## destination port

To set the destination port of the rule, use the following command:

```
destination port <NUMBER/WORD>
```

## Syntax Description

---

destination port	Set the destination port of the rule
<NUMBER/WORD>	Set the destination to this port number

---

### Defaults

None.

### Example

```
ruckus(config-l3acl-rule)# destination port 580
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### protocol

To set the protocol for the rule, use the following command:

```
protocol <NUMBER/WORD>
```

## Syntax Description

---

protocol	Set the protocol for the rule
<NUMBER/WORD>	Set to this protocol

---

### Defaults

None.

### Example

```
ruckus(config-l3acl-rule)# protocol tcp
```

The protocol must be a number between 0 and 254.

```
ruckus(config-l3acl-rule)# protocol Any
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### show

To display L3/L4/IP ACL settings, use the following command:

```
show
```

### **Example**

```
ruckus(config-l3acl) # show
L3/L4/IP ACL:
  ID:
  :
  Name= l3acl1
  Description=
  Default Action if no rule is matched= Deny all by default
  Rules:
    1:
      Description=
      Type= Allow
      Destination Address= 192.168.1.22/24
      Destination Port= 53
      Protocol= Any
    2:
      Description=
      Type= Allow
      Destination Address= Any
      Destination Port= 67
      Protocol= Any

ruckus(config-l3acl) #
```

## **Layer 3 IPv6 Access Control List Commands**

Use the `l3acl-ipv6` command to configure the IPv6 Layer 3/Layer 4/IP Access Control List. To run these commands, you must first enter the `config-l3acl` context.

### **l3acl-ipv6**

To enter the `config-l3acl-ipv6` context, run this command:

```
l3acl-ipv6 <WORD>
```

## **abort**

Exits the config-l3acl-ipv6 context without saving changes.

## **end**

Saves changes, and then exits the config-l3acl-ipv6 context.

## **exit**

Saves changes, and then exits the config-l3acl-ipv6 context.

## **quit**

Exits the config-l3acl-ipv6 context without saving changes.

## **name**

Sets the L3/L4/IPv6 ACL entry name.

## **description**

Sets the L3/L4/IPv6 ACL entry description.

## **mode allow**

Sets the ACL mode to 'allow'.

## **mode deny**

Sets the ACL mode to 'deny'.

## **no rule-order**

Deletes a rule name from the L3/L4/IPv6 ACL.

## **rule-order**

Creates a new L3/L4/IPv6 ACL rule or modifies an existing entry rule.

## **Configure L3 IPv6 Rule Commands**

Use the `l3acl-ipv6-rule` commands to configure the IPv6 Layer 3/Layer 4/IP Access Control List rules. To run these commands, you must first enter the `config-l3acl-ipv6-rule` context. To enter the `config-l3acl-ipv6-rule` context, run this command:

```
rule-order <NUMBER>
```

### **end**

Saves changes, and then exits the config-l3acl-ipv6-rule context.

### **exit**

Saves changes, and then exits the config-l3acl-ipv6-rule context.

### **order**

Sets the L3/L4/IPv6 ACL rule order.

### **description**

Sets the L3/L4/IPv6 ACL rule description.

### **type allow**

Sets the ACL rule type to 'allow'.

### **type deny**

Sets the ACL rule type to 'deny'.

### **destination**

Contains commands that can be executed from within the context.

### **destination address**

Sets the destination address of a L3/L4/IPv6 ACL rule.

### **destination port**

Sets the destination port of a L3/L4/IPv6 ACL rule.

### **protocol**

Sets the protocol of a L3/L4/IPv6 ACL rule.

### **icmpv6-type Any**

Sets the icmpv6 type of a L3/L4/IPv6 ACL rule.



## **icmpv6-type number**

Sets the icmpv6 type of a L3/L4/IPv6 ACL rule.

## **show**

Displays L3/L4/IPv6 ACL settings.

# **Configure Precedence Policy Commands**

Use the `prece` commands to configure precedence policy settings. Precedence policies are used to define the order in which VLAN and rate limiting policies are applied when the WLAN settings, AAA server configuration or Device Policy settings conflict.

To run these commands, you must first enter the `config-prece` context.

## **prece**

To create or modify a precedence policy, use the following command:

```
prece <WORD>
```

Enters the `config-prece` context. To save changes and exit the context, type `exit` or `end`. To exit the context without saving changes, type `abort`.

## **Example**

```
ruckus(config)# prece precedence1  
The Precedence Policy entry 'precedence1' has been created.  
ruckus(config-prece) #
```

## **name**

Sets the Precedence Policy entry name.

## **description**

Sets the Precedence Policy entry description.

# **Configure Precedence Policy Rule Commands**

Use the following commands to configure precedence policy rules.

## rule

Creates a new Precedence Policy rule or modifies an existing entry rule. Enters the `config-prece-rule` context.

```
rule <NUMBER>
```

### Syntax Description

<code>rule</code>	Create a rule and enter the rule creation context.
<code>&lt;NUMBER&gt;</code>	Enter the rule number (1-2). Each precedence policy can have up to two rules.
<code>description</code>	Sets the Precedence Policy rule description.
<code>order &lt;WORD&gt;</code>	Sets the order of a Precedence Policy rule. The default order is AAA, Device Policy, WLAN.
<code>show</code>	Displays precedence policy settings.

### Example

```
ruckus(config)# prece precedencel
```

The Precedence Policy entry 'precedencel' has been created.

```
ruckus(config-prece)# rule 1
```

```
ruckus(config-prece-rule)# order "Device Policy" "WLAN" "AAA"
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-prece-rule)# end
```

```
ruckus(config-prece)# show
```

Precedence Policy:

```
  ID:
```

```
  :
```

```
    Name= precedencel
```

```
    Description=
```

```
    Rules:
```

```
      1:
```

```
        Description=
```

```
        Attribute = vlan
```

```
        Order = Device Policy,WLAN,AAA
```

```
      2:
```

```
        Description=
```

```
Attribute = rate-limit
Order = AAA,Device Policy,WLAN
```

```
ruckus(config-prece)#
ruckus(config-prece)# end
The Precedence Policy entry has saved successfully.
Your changes have been saved.
```

## no prece

To delete a precedence policy entry, use the following command:

```
no prece <WORD>
```

# Configure Device Policy Commands

Use the device policy commands to configure access control and rate limiting policies based on client type. To run these commands, you must first enter the `config-dvc-psy` context.

## dvcpsy

To create a device policy or edit an existing device policy, enter the following command:

```
dvcpsy <WORD>
```

## Syntax Description

show	Display device policy settings.
name <WORD>	Set the device policy entry name.
description <WORD>	Sets the device policy entry description.
mode <WORD>	Sets the device policy entry default mode (allow or deny).
no <NUMBER>	Delete a rule.
rule <NUMBER>	Create or modify a rule. Enter the <code>config-dvc-psy-rule</code> context. You can create up to nine rules per access policy (one for each OS/Type).

## Defaults

None.

## Example

```
ruckus(config)# dvcpcy devpcy1
```

The Device Policy entry 'devpcy1' has been loaded. To save the Device Policy entry, type `end` or `exit`.

```
ruckus(config-dvc-pcy)# name device_policy_1
```

The command was executed successfully. To save the changes, type `'end'` or `'exit'`.

```
ruckus(config-dvc-pcy)# description "deny iOS"
```

The command was executed successfully. To save the changes, type `'end'` or `'exit'`.

```
ruckus(config-dvc-pcy)# rule 1
```

```
ruckus(config-dvc-pcy-rule)# type deny
```

The command was executed successfully. To save the changes, type `'end'` or `'exit'`.

```
ruckus(config-dvc-pcy-rule)# devinfo "Apple iOS"
```

The command was executed successfully. To save the changes, type `'end'` or `'exit'`.

```
ruckus(config-dvc-pcy-rule)# vlan none
```

The command was executed successfully. To save the changes, type `'end'` or `'exit'`.

```
ruckus(config-dvc-pcy-rule)# rate-limit uplink 10 downlink 10
```

The command was executed successfully. To save the changes, type `'end'` or `'exit'`.

```
ruckus(config-dvc-pcy-rule)# end
```

```
ruckus(config-dvc-pcy)# show
```

Device Policy:

ID:

1:

Name= device\_policy\_1

Description= deny iOS

Default Mode= deny

Rules:

1:

Description=

```
OS/Type = Apple iOS
Type= deny
VLAN = Any
Rate Limiting Uplink = 10.00Mbps
Rate Limiting Downlink = 10.00Mbps
```

```
ruckus(config-dvc-pcy)# end
```

The Device Policy entry has saved successfully.

Your changes have been saved.

```
ruckus(config)# show dvcpcy
```

Device Policy:

ID:

2:

Name= device\_policy\_1

Description= deny iOS

Default Mode= deny

Rules:

1:

Description=

OS/Type = Apple iOS

Type= deny

VLAN = Any

Rate Limiting Uplink = 10.00Mbps

Rate Limiting Downlink = 10.00Mbps

```
ruckus(config)#
```

## rule

Use the rule command from within the config-dvc-pcy context to create or edit a device policy rule and enter the config-dvc-pcy-rule context. Up to 9 rules can be created per device policy.

### **Syntax Description**

---

rule	Create or edit a device policy rule. Enter the config-dvc-pcy-rule context.
------	---

---

description <WORD>	Set the Device Policy rule description.
devinfo <WORD>	Set the operating system type of a device policy rule.
type <WORD>	Set the device policy rule type (allow or deny).
vlan <NUMBER>	Set the VLAN ID to the number specified or "none."
rate-limit uplink <NUMBER> downlink <NUMBER>	Set the rate limiting uplink and downlink speeds in mbps.
no rate-limit	Set rate limiting to disabled.

### **Example**

```
ruckus(config-dvc-pcy)# rule 2
ruckus(config-dvc-pcy-rule)# description "rate limit gaming
devices"
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-dvc-pcy-rule)# devinfo "Gaming"
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-dvc-pcy-rule)# type allow
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-dvc-pcy-rule)# vlan none
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-dvc-pcy-rule)# rate-limit uplink 0.1 downlink 0.1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-dvc-pcy-rule)# end
ruckus(config-dvc-pcy)# show
Device Policy:
  ID:
    2:
      Name= device_policy_1
      Description= deny iOS
      Default Mode= deny
      Rules:
```

```
1:
  Description=
  OS/Type = Apple iOS
  Type= deny
  VLAN = Any
  Rate Limiting Uplink = 10.00Mbps
  Rate Limiting Downlink = 10.00Mbps
2:
  Description= rate limit gaming devices
  OS/Type = Gaming
  Type= allow
  VLAN = Any
  Rate Limiting Uplink = 0.10Mbps
  Rate Limiting Downlink = 0.10Mbps
```

```
ruckus(config-dvc-pcy) #
```

## **no dvcpcy**

To delete a device policy, use the following command:

```
no dvcpcy <WORD>
```

# **Configure Application Denial Policy Commands**

Use the following commands to create or modify application denial policies.

## **app-denial-policy**

To create a new application policy or modify an existing policy, use the following command:

```
app-denial-policy <WORD>
```

## ***Syntax Description***

---

abort	Exits the config-app-denial-policy context without saving changes.
end	Saves changes, and then exits the config-app-denial-policy context.

---

---

exit	Saves changes, and then exits the config-app-denial-policy context.
quit	Exits the config-app-denial-policy context without saving changes.
show	Displays Application Denial Policy settings.
name <WORD>	Sets the Application Denial Policy entry name.
description <WORD>	Sets the Application Denial Policy entry description.
no rule <NUMBER>	Deletes a rule name.
rule <NUMBER>	Creates a new Application Denial Policy rule or modifies an existing entry.

---

### **Example**

```
ruckus(config)# app-denial-policy policy1
The Application Denial Policy entry 'policy1' has been created.
ruckus(config-app-denial-policy)# rule 1
ruckus(config-app-denial-policy-rule)# application HTTP hostname
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-app-denial-policy-rule)# description facebook.com
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-app-denial-policy-rule)# end
ruckus(config-app-denial-policy)# end
The Application Denial Policy entry has saved successfully.
Your changes have been saved.
ruckus(config)# show app-denial-policy
Application Denial Policy:
  ID:
  1:
    Name= policy1
    Description=
    Default Mode= accept
    Rules:
      1:
        Application= HTTP hostname
        Description= facebook.com
```



```
ruckus(config)#
```

## **no app-denial-policy**

To delete an Application Denial Policy entry, use the following command:

```
no app-denial-policy <WORD>
```

### **Example**

```
ruckus(config)# no app-denial-policy policy1
```

The Application Denial Policy 'policy1' has been deleted.

```
ruckus(config)#
```

## **Configure Application Denial Policy Rules**

Use the following commands to configure application denial policy rules.

### **no rule**

To delete a rule, use the following command:

```
no rule <NUMBER>
```

### **rule**

Creates a new Application Denial Policy rule or modifies an existing entry. Enters the config-app-denial-policy-rule context.

```
rule <NUMBER>
```

### **Syntax Description**

abort	Exits the config-app-denial-policy-rule context without saving changes.
end	Saves changes, and then exits the config-app-denial-policy-rule context.
exit	Saves changes, and then exits the config-app-denial-policy-rule context.
quit	Exits the config-app-denial-policy-rule context without saving changes.

---

application <WORD>	Sets the application of Application Denial Policy rule.
description <WORD>	Sets the description of Application Denial Policy rule.

---

## **Defaults**

None

## **Example**

```
ruckus(config)# app-denial-policy policy1
The Application Denial Policy entry 'policy1' has been loaded. To
save the Application Denail Policy entry, type end or exit.
ruckus(config-app-denial-policy)# rule 1
ruckus(config-app-denial-policy-rule)# application "HTTP hostname"
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-app-denial-policy-rule)# description facebook.com
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-app-denial-policy-rule)# end
ruckus(config-app-denial-policy)# show
Application Denial Policy:
  ID:
    1:
      Name= policy1
      Description=
      Default Mode= accept
      Rules:
        1:
          Application= HTTP hostname
          Description= facebook.com

ruckus(config-app-denial-policy)#
```

# Configuring User-Defined Applications

Use the following commands to configure user-defined applications. Once created, user-defined applications can be blocked using the application denial policy commands.

## user-defined-app

To configure User Defined Application settings, and enter the config-user-defined-app context, use the following command:

```
user-defined-app
```

### Example

```
ruckus(config)# user-defined-app
ruckus(config-user-defined-app)# rule rule1
The User Defined Application entry rule1 has been created.
ruckus(config-user-defined-app-rule)# application skype
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-user-defined-app-rule)# destination-IP 192.168.10.4
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-user-defined-app-rule)# netmask 255.255.255.0
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-user-defined-app-rule)# destination-port 100
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-user-defined-app-rule)# end
ruckus(config-user-defined-app)# show
User Defined Application:
  ID:
    1:
      Application= skype
      DST-IP= 192.168.10.4
      Netmask= 255.255.255.0
      DST-Port= 100
      Protocol=

ruckus(config-user-defined-app)#
```

## **exit**

Saves changes, and then exits the config-user-defined-app context.

## **end**

Saves changes, and then exits the config-user-defined-app context.

## **show**

Displays User defined Application settings.

## **no rule**

Deletes an User Defined Application.

```
no rule <WORD>
```

## **rule**

Creates a new User defined Application rule or modifies an existing entry. Enters the config-user-defined-app-rule context.

```
rule <WORD>
```

## **abort**

Exits the config-user-defined-app-rule context without saving changes.

## **end**

Saves changes, and then exits the config-user-defined-app-rule context.

## **exit**

Saves changes, and then exits the config-user-defined-app-rule context.

## **destination-IP**

Sets the destination address of a User defined Application rule.

```
destination-IP <IP-ADDR>
```

## **netmask**

Sets the netmask of a User defined Application rule.

```
netmask <IP-ADDR>
```

## **destination-port**

Sets the destination port of a User defined Application rule.

```
destination-port <NUMBER>
```

## **protocol**

Sets the protocol of a User defined Application rule.

```
protocol <WORD>
```

## **application**

Sets the application of User defined Application rule.

```
application <WORD>
```

# **Configure Application Port Mapping**

Use the following commands to configure application port mapping.

## **app-port-mapping**

Configures Application Port Map settings. Enters config-app-port-mapping context.

## **exit**

Saves changes, and then exits the config-app-port-mapping context.

## **end**

Saves changes, and then exits the config-app-port-mapping context.

## **show**

Displays Application Port Mapping settings.

## **no rule**

Deletes an Application Port Mapping rule.

```
no rule <WORD>
```

## **rule**

Creates a new Application Port Mapping rule or modifies an existing entry. Enters config-app-port-mapping-rule context.

```
rule <WORD>
```

**abort**

Exits the config-app-port-mapping context without saving changes.

**end**

Saves changes, and then exits the config-app-port-mapping context.

**exit**

Saves changes, and then exits the config-app-port-mapping context.

**port**

Sets the Port of Application Port Mapping rule.

```
port <NUMBER>
```

**description**

Sets the Description of Application Port Mapping rule.

```
description <WORD>
```

**protocol**

Sets the Protocol of Application Port Mapping rule.

```
protocol <WORD>
```

## Configure Whitelist Commands

Use the `whitelist` command to create a new client isolation whitelist or modify an existing whitelist, and enter the `config-whitelist` context.

**whitelist**

To create a new white list entry or modify an existing entry, use the following command:

```
whitelist <WORD>
```

**no whitelist**

To delete a whitelist entry, use the following command:

```
no whitelist <WORD>
```

**name**

To set the White List entry name, use the following command:

```
name <WORD>
```

**description**

To set the description of the whitelist entry, use the following command:

```
description <WORD>
```

**Configuring Whitelist Rules**

Use the `rule` command from within the `config-whitelist` context to create a new rule or modify an existing rule, and enter the `config-whitelist-rule` context.

**rule**

To create a new whitelist rule or modify an existing rule, use the following command:

```
rule <NUMBER>
```

**no rule**

To delete a whitelist rule, use the following command:

```
no rule <NUMBER>
```

**description**

To set the White List rule description, use the following command:

```
description <WORD>
```

**mac**

To set the MAC address, use the following command (format: XX:XX:XX:XX:XX:XX):

```
mac <MAC>
```

**ip**

To set the IP address, use the following command (format: 172.18.110.12).

```
ip <IP>
```

# Configure Band Balancing Commands

Client Band Balancing attempts to balance the number of clients across AP radios, allowing configurable thresholds for ratio of clients on the 2.4 vs. 5 GHz radio bands. Use the band-balancing commands to configure the controller's band balancing settings. To run these commands, you must first enter the `config-band-balancing` context.

## **band-balancing**

To enable load-balancing and enter the `config-band-balancing` context, use the following command:

```
band-balancing
```

## **abort**

Exits the band balancing context without saving changes.

## **end**

Saves changes, and then exits the band balancing context.

## **exit**

Saves changes, and then exits the band balancing context.

## **quit**

Exits the band balancing context without saving changes.

## **enable**

Enable the band balancing settings.

## **disable**

Disables the band balancing settings.

## **percent-2.4G <NUMBER>**

Configures percent of clients on 2.4G band.

## **show**

Displays information about Band balancing.



### ***Example***

```
ruckus(config)# band-balancing
ruckus(config-band-balancing)# enable
The band balancing settings have been updated.
ruckus(config-band-balancing)# percent-2.4G 25
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-band-balancing)# show
Band Balancing:
  Status= Enabled
  Percent of clients on 2.4G band: 25%

ruckus(config-band-balancing)# end
The band balancing settings have been updated.
ruckus(config)#
```

## **Configure Load Balancing Commands**

Client Load Balancing attempts to balance the number of clients across APs, per radio band. Use the load-balancing commands to configure the controller's load balancing settings. To run these commands, you must first enter the `config-load-balancing` context.

### **load-balancing**

To enable load-balancing and enter the `config-load-balancing` context, use the following command:

```
load-balancing
```

### ***Example***

```
ruckus(config)# load-balancing
ruckus(config-load-balancing)#
```

### **no load-balancing**

To disable load balancing settings (from the `config` context), use the following command:

```
no load-balancing
```

### **Example**

```
ruckus(config)# no load-balancing  
The load balancing settings have been updated.  
ruckus(config)# show load-balancing
```

```
Load Balancing:  
  Status= Disabled  
  Radio:  
    0:  
      AdjacentThreshold= 50  
      WeakBypass= 33  
      StrongBypass= 55  
      ActivationThreshold= 10  
      NewTrigger= 3  
      Headroom= 3  
  
    1:  
      AdjacentThreshold= 43  
      WeakBypass= 35  
      StrongBypass= 55  
      ActivationThreshold= 10  
      NewTrigger= 3  
      Headroom= 3
```

```
ruckus(config)#
```

### **adj-threshold**

To configure the adjacent threshold for load balancing, use the following command:

```
adj-threshold [wifi0|wifi1] <NUMBER>
```

### **Syntax Description**

---

adj-threshold	Configure the adjacent threshold for load balancing
---------------	---

---

---

wifi0, wifi1	Configure this interface
<NUMBER>	Set the adjacent threshold value (1~100)

---

### **Defaults**

Wifi0: 50

Wifi1: 43

### **weak-bypass**

To configure the weak bypass for load balancing, use the following command:

```
weak-bypass [wifi0|wifi1] <NUMBER>
```

### **Syntax Description**

---

weak-bypass	Configure the weak bypass for load balancing
wifi0, wifi1	Configure this interface
<NUMBER>	Set the weak-bypass value (1~100)

---

### **Defaults**

wifi0: 33

wifi1: 35

### **strong-bypass**

To configure the strong bypass for load balancing, use the following command:

```
strong-bypass [wifi0|wifi1] <NUMBER>
```

### **Syntax Description**

---

strong-bypass	Configure the strong bypass for load balancing
wifi0, wifi1	Configure this interface
<NUMBER>	Set the strong-bypass value (1~100)

---

### **Defaults**

55

## act-threshold

To configure the activation threshold for load balancing, use the following command:

```
act-threshold [wifi0|wifi1] <NUMBER>
```

### **Syntax Description**

act-threshold	Configure the activation threshold for load balancing.
wifi0, wifi1	Configure this interface.
<NUMBER>	Set the activation threshold value (1~100).

### **Defaults**

10

### **Example**

```
ruckus(config-load-balancing)# act-threshold wifi0 50
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-load-balancing)#
```

## new-trigger

To configure new trigger threshold (1-100), use the following command:

```
new-trigger [wifi0|wifi1] <NUMBER>
```

### **Syntax Description**

new-trigger	Configure a new trigger threshold for the specified interface.
wifi0, wifi1	Configure this interface.
<NUMBER>	Set the new trigger threshold value (1~100).

### **Defaults**

3

## Example

```
ruckus(config-load-balancing)# new-trigger wifi0 3
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-load-balancing)#
```

## headroom

To configure headroom settings for the specified interface, use the following command:

```
headroom [wifi0|wifi1] <NUMBER>
```

## Syntax Description

headroom	Configure headroom for the specified interface.
wifi0, wifi1	Configure this interface.
<NUMBER>	Set the headroom value (1~100).

## Defaults

3

## Example

```
ruckus(config-load-balancing)# headroom wifi0 3
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-load-balancing)#
```

## disable wifi0

Disable wifi0 load balancing.

## disable wifi1

Disable wifi1 load balancing.

## enable wifi0

Enable wifi0 load balancing.

## **enable wifi1**

Enable wifi1 load balancing.

## **show**

To display the current service settings, use the following command:

```
show
```

## ***Syntax Description***

---

show	Display the current service settings
------	--------------------------------------

---

## ***Defaults***

None.

## ***Example***

```
ruckus(config-load-balancing)# show
Load Balancing:
  Status= Disabled
  Radio:
    0:
      AdjacentThreshold= 50
      WeakBypass= 33
      StrongBypass= 55
      ActivationThreshold= 1
      NewTrigger= 3
      Headroom= 3
    1:
      AdjacentThreshold= 43
      WeakBypass= 35
      StrongBypass= 55
      ActivationThreshold= 10
      NewTrigger= 3
      Headroom= 3
```

```
ruckus(config-load-balancing)#
```

## Configure STP Commands

Both Ethernet ports of a ZoneDirector 1000/1100/3000/5000 are one Logical interface. They are designed to provide high availability connections to separate switches and do not provide dual-port ISL channel bonding. Switches should use STP to block one path. The default for Zonedirector is “no stp”.

### **stp**

To enable Spanning Tree Protocol, use the following command:

```
stp
```

### **no stp**

To disable Spanning Tree Protocol, use the following:

```
no stp
```

## Configure System Commands

Use the `sys` or `system` command to configure the controller’s system settings, including its host name, FlexMaster server, NTP server, SNMP, and QoS settings. To run these commands, you must first enter the `config-sys` context.

### **system**

To enter the `config-sys` context and configure system settings, use the following command:

```
system
```

### **Example**

```
ruckus(config)# system  
ruckus(config-sys)#
```

### **dot11-country-code**

To set the controller’s country code, use the following command:

```
dot11-country-code <COUNTRY-CODE> {arguments}
```

## Syntax Description

<code>dot11-country-code</code>	Configure the controller's country code setting
<code>&lt;COUNTRY-CODE&gt;</code>	Set the country code to this value
<code>channel-mode</code>	Contains commands that can be executed from within the context
<code>allow-indoor</code>	Allows ZoneFlex Outdoor APs to use channels regulated as indoor use-only
<code>not-allow-indoor</code>	Disallows ZoneFlex Outdoor APs to use channels regulated as indoor use-only
<code>channel-optimization</code>	Set channel optimization type (compatibility, interoperability, performance)

## Defaults

None.

## Example

To set the country code to US, enter the following command:

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# dot11-country-code US
The country code settings have been updated.
ruckus(config-sys)#
```

## hostname

To set the system hostname, use the following command:

```
hostname
```

## Syntax Description

<code>hostname</code>	Set the controller's system hostname
-----------------------	--------------------------------------

## Defaults



None

### **Example**

```
ruckus(config-sys)# hostname ruckus-xjoe
```

The system identity/hostname settings have been updated.

## **Interface Commands**

Use the `interface` commands to configure the controller's IP address and VLAN settings. To run these commands, you must first enter the `config-sys-if` context.

### **interface**

To enter the `config-sys-if` context and configure IP address and VLAN settings, use the following command:

```
interface
```

### **Example**

```
ruckus(config-sys)# interface
```

```
ruckus(config-sys-if)#
```

### **ip enable**

To enable IPv4 addressing, use the following command:

```
ip enable
```

### **ip route gateway**

To set the controller's gateway IP address, use the following command:

```
ip route gateway <GATEWAY-ADDR>
```

### **Syntax Description**

---

<code>ip route gateway</code>	Configure the controller's gateway IP address
<code>&lt;GATEWAY-ADDR&gt;</code>	Set the controller' gateway IP address to this value

---

### **Defaults**

None.

### **Example**

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# interface
ruckus(config-sys-if)# ip route gateway 192.168.0.1
```

The command was executed successfully.

### **ip name-server**

To set the controller's DNS servers, use the `ip name-server` command. Use a space to separate the primary and secondary DNS servers.

```
ip name-server <DNS-ADDR> [<DNS-ADDR>]
```

### **Syntax Description**

<code>ip name-server</code>	Configure the controller's DNS server address or addresses
<code>DNS-ADDR</code>	Set the DNS server address to this value. If entering primary and secondary DNS server addresses, use a space to separate the two addresses.

### **Defaults**

192.168.0.1

### **Example**

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# interface
ruckus(config-sys-if)# ip name-server 192.168.0.1
```

The command was executed successfully.

### **ip addr**

To set the controller's IP address and netmask, use the following command:

```
ip addr <IP-ADDR> <NET-MASK>
```

Use a space to separate the IP address and netmask.

## Syntax Description

<code>ip addr</code>	Configure the controller's IP address and netmask
<code>&lt;IP-ADDR&gt;</code>	Set the controller's IP address to this value
<code>&lt;NET-MASK&gt;</code>	Set the controller's netmask to this value

## Defaults

IP address: 192.168.0.2

Subnet mask: 255.255.255.0

## Example

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# interface
ruckus(config-sys-if)# ip addr 192.168.0.2 255.255.255.0
The command was executed successfully.
```

## ip mode

To set the controller's IP address mode, use the following command:

```
ip mode <dhcp|static>
```

## Syntax Description

<code>ip mode</code>	Configure the controller's IP address mode
<code>dhcp</code>	Set the controller's IP address mode to DHCP
<code>static</code>	Set the controller's IP address mode to static

## Defaults

None.

## Example

To set the controller's IP address mode to DHCP, enter the following command:

```
ruckus# config
```

```
ruckus(config)# system
ruckus(config-sys)# interface
ruckus(config-sys-if)# ip mode dhcp
The command was executed successfully.
```

## show

To display the current management interface settings, use the following command:

```
show
```

## Syntax Description

---

show	Display the current management interface settings
------	---

---

## Defaults

None.

## Example

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# interface
ruckus(config-sys-if)# show
Protocol Mode= IPv4-Only
Device IP Address:
Mode= Manual
IP Address= 192.168.11.100
Netmask= 255.255.255.0
Gateway Address= 192.168.11.1
Primary DNS= 192.168.11.1
Secondary DNS= 168.95.1.1

Management VLAN:
Status= Disabled
VLAN ID=

ruckus(config-sys-if)#
```

## ipv6 enable

To enable IPv6 addressing, use the following command:

```
ipv6 enable
```

## ipv6 route gateway

To set the controller's IPv6 gateway addressing, use the following command:

```
ipv6 route gateway <GATEWAY-ADDR>
```

## ipv6 name-server

To set the IPv6 DNS server, use the following command:

```
name-server <DNS-ADDR> [<DNS-ADDR>]
```

## ipv6 addr

To set the IPv6 addressing, use the following command:

```
addr <IPv6-ADDR> <IPv6-PREFIX>
```

## ipv6 mode

To set the IPv6 address mode, use the following command:

```
ipv6 mode [auto|manual]
```

## vlan

If the ZoneDirector is on a tagged Access VLAN, to set the VLAN ID, use the following command:

```
vlan <NUMBER>
```

## no ip

To disable IPv4 addressing, use the following command:

```
no ip
```

## no ipv6

To disable IPv6 addressing, use the following command:

```
no ipv6
```

## no ntp

To disable the NTP client, use the following command:

```
no ntp
```

### **Syntax Description**

---

<code>no ntp</code>	Disable the NTP client on the controller.
---------------------	---

---

### **Defaults**

Enabled. The default NTP server addresss is `ntp.ruckuswireless.com`.

### **Example**

```
ruckus(config-sys)# no ntp
The NTP settings have been updated.
```

### **ntp**

To enable the NTP client, use the following command:

```
ntp <IP-ADDR/DOMAIN-NAME>
```

### **Syntax Description**

---

<code>ntp</code>	Enable the NTP client
<code>&lt;IP-ADDR/ DOMAIN-NAME&gt;</code>	Set the NTP server address to this IP address/domain name

---

### **Defaults**

None.

### **Example**

```
ruckus(config-sys)# ntp 192.168.2.21
The NTP settings have been updated.
ruckus(config-sys)# ntp sohu.com
The NTP settings have been updated.
```

### **ftp-anon**

To enable FTP anonymous access, use the following command:

```
ftp-anon
```

### **no ftp-anon**

To disable FTP anonymous access, use the following command:

```
no ftp-anon
```

### **ftp**

Enable FTP server.

### **no ftp**

Disable FTP server.

## **Smart Redundancy Commands**

To configure the Smart Redundancy settings, you must first enter the `config-sys-smart-redundancy` context from within the `config-sys` context.

### **smart-redundancy**

To enter the `config-sys-smart-redundancy` context and configure Smart Redundancy settings, use the following command:

```
smart-redundancy
```

### ***Syntax Description***

<code>smart-redundancy</code>	Configures smart redundancy settings.
<code>abort</code>	Exits the smart redundancy context without saving changes.
<code>end</code>	Saves changes, and then exits the smart redundancy context.
<code>exit</code>	Saves changes, and then exits the smart redundancy context.
<code>quit</code>	Exits the smart redundancy context without saving changes
<code>peer-addr &lt;IP-ADDR&gt;</code>	Sets the peer's IP/IPv6 address.
<code>secret &lt;WORD&gt;</code>	Sets the shared secret to the specified secret.

---

show	Displays information about smart redundancy.
------	--

---

### **Example**

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# smart-redundancy
ruckus(config-sys-smart-redundancy)# peer-addr 192.168.40.101
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-sys-smart-redundancy)# secret secret
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-sys-smart-redundancy)# show
Smart Redundancy:
  Status= Disabled
  Peer IP/IPv6 Address=
  Shared Secret=

ruckus(config-sys-smart-redundancy)# end
The smart redundancy settings have been updated.
Your changes have been saved.
ruckus(config-sys)#
```

### **no smart-redundancy**

Disables the smart redundancy settings.

### **Example**

```
ruckus(config-sys)# no smart-redundancy
The smart redundancy settings have been updated.
ruckus(config-sys)#
```

## **Management Interface Commands**

To configure management interface settings, you must first enter the config-sys-mgmt-if context from the config-sys context.



## mgmt-if

To enter the `config-sys-mgmt-if` context and configure the management interface settings, use the following command:

```
mgmt-if
```

### *Syntax Description*

---

mgmt-if	Configure the management interface settings
---------	---

---

### *Defaults*

None.

### *Example*

```
ruckus(config-sys) # mgmt-if  
ruckus(config-sys-mgmt-if) #
```

## no mgmt-if

To disable the management interface, use the following command:

```
no mgmt-if
```

### *Syntax Description*

---

no mgmt-if	Disable the management interface
------------	----------------------------------

---

### *Defaults*

None.

### *Example*

```
ruckus(config-sys) # no mgmt-if  
The management interface has been updated.
```

## ip addr

To set the management interface IP address, use the following command:

```
ip addr <IP-ADDR> <NET-MASK>
```

## gateway

To set the management interface gateway address, use the following command:

```
gateway <GATEWAY-ADDR>
```

## no gateway

To disable the management interface gateway address, use the following command:

```
no gateway
```

## vlan

To enable the management VLAN and set the VLAN ID, use the following command:

```
vlan <NUMBER>
```

## mgmt-if-ipv6

To enter the `config-sys-mgmt-if-ipv6` context and configure the management interface settings, use the following command:

```
mgmt-if-ipv6
```

## Syntax Description

---

mgmt-if-ipv6	Configure the management interface settings
--------------	---

---

## Defaults

None.

## Example

```
ruckus(config-sys) # mgmt-if-ipv6  
ruckus(config-sys-mgmt-if-ipv6) #
```

## no mgmt-if-ipv6

To disable the management interface, use the following command:

```
no mgmt-if-ipv6
```

## Syntax Description

---

<code>no mgmt-if-ipv6</code>	Disable the management interface
------------------------------	----------------------------------

---

### Defaults

None.

### Example

```
ruckus(config-sys) # no mgmt-if-ipv6
```

The management interface has been updated.

### ipv6 addr

To set the management interface IP address, use the following command:

```
ip addr <IPv6-ADDR> <IPv6-PREFIX>
```

### gateway

To set the management interface gateway address, use the following command:

```
gateway <GATEWAY-ADDR>
```

### no gateway

To disable the management interface gateway address, use the following command:

```
no gateway
```

### vlan

To enable the management VLAN and set the VLAN ID, use the following command:

```
vlan <NUMBER>
```

### flexmaster

To set the FlexMaster server address and the periodic inform interval, use the following command:

```
flexmaster <IP-ADDR/DOMAIN-NAME> interval <NUMBER>
```

## Syntax Description

flexmaster	Configure the FlexMaster server settings
<IP-ADDR/DOMAIN-NAME>	Set to this URL or IP address
interval	Configure the periodic inform interval
<NUMBER>	Set to this interval (in minutes)

## Defaults

None.

## Example

```
ruckus(config-sys)# flexmaster http://172.18.30.118 interval 30
The FlexMaster Management settings have been updated.
```

## no flexmaster

To disable FlexMaster management of the controller, use the following command:

```
no flexmaster
```

## Syntax Description

no flexmaster	Disable FlexMaster management of the controller
---------------	---

## Defaults

None

## Example

```
ruckus(config-sys)# no flexmaster
FlexMaster Management has been disabled.
```

## northbound

To enable northbound portal interface support and set the northbound portal password, use the following command:

```
northbound password <WORD>
```

## Defaults

Disabled

## Example

```
ruckus(config-sys)# northbound password pass123
```

The northbound portal interface settings have been updated.

## no northbound

To disable northbound portal interface support, use the following command:

```
no northbound
```

## Example

```
ruckus(config-sys)# no northbound
```

Northbound portal interface has been disabled.

## SNMPv2 Commands

Use the following commands to configure SNMPv2 settings. To use these commands, you must first enter the `config-sys-snmpv2` context.

### snmpv2

To configure the SNMPv2 settings, use the following command:

```
snmpv2
```

Executing this command enters the `config-sys-snmpv2` context.

### Syntax Description

<code>snmpv2</code>	Configure the SNMPv2 settings
<code>abort</code>	Exits the <code>config-sys-snmpv2</code> context without saving changes.
<code>end</code>	Saves changes, and then exits the <code>config-sys-snmpv2</code> context.
<code>exit</code>	Saves changes, and then exits the <code>config-sys-snmpv2</code> context.

quit	Exits the config-sys-snmpv2 context without saving changes.
no access-v3	Disables special MIB node for customer's kt.
access-v3	Enables special MIB node for customer's kt.
contact <WORD>	Enables SNMPV2 agent and sets the system contact.
location <WORD>	Enables SNMPV2 agent and sets the system location.
ro-community <WORD>	Enables SNMPV2 agent and sets the RO community name.
rw-community <WORD>	Enables SNMPV2 agent and sets the RW community name.
show	Displays SNMPV2 agent and SNMP trap settings.

### **Defaults**

SNMP Agent:

```
Status= Enabled
Contact= https://support.ruckuswireless.com/contact_us
Location= 350 West Java Dr. Sunnyvale, CA 94089 US
RO Community= public
RW Community= private
```

SNMP Trap:

```
Format= Version2
Status= Disabled
```

Support-access-V3:

```
Status= Disabled
```

### **Example**

```
ruckus(config-sys)# snmpv2
ruckus(config-sys-snmpv2)#
```

### **contact**

To enable SNMPV2 agent and set the system contact, use the following command:

```
contact <WORD>
```

## location

To enable SNMPv2 agent and set the system location, use the following command:

```
location <WORD>
```

## ro-community

To set the read-only (RO) community name, use the following command:

```
ro-community <WORD>
```

### *Syntax Description*

ro-community	Configure the read-only community name
<WORD>	Set the read-only community name to this value

### *Defaults*

public

### *Example*

```
ruckus(config-sys-snmpv2)# ro-community private-123
```

The command was executed successfully

## rw-community

To set the read-write (RW) community name, use the following command:

```
rw-community <WORD>
```

This command must be entered from within the `snmp-agent` context.

### *Syntax Description*

rw-community	Configure the read-write community name
<WORD>	Set the read-write community name to this value

### *Defaults*

private

### *Example*

```
ruckus(config-sys-snmpv2)# rw-community private-123
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## show

To display SNMPv2 agent and SNMP trap settings, use the show command.

### Example

```
ruckus(config-sys-snmpv2)# show
```

SNMP Agent:

```
Status= Enabled
```

```
Contact= https://support.ruckuswireless.com/contact_us
```

```
Location= 350 West Java Dr. Sunnyvale, CA 94089 US
```

```
RO Community= public
```

```
RW Community= private
```

SNMP Trap:

```
Format= Version2
```

```
Status= Disabled
```

Support-access-V3:

```
Status= Disabled
```

## snmpv2-ap

To enable SNMP AP notification, use the following command:

```
snmpv2-ap
```

### Example

```
ruckus(config-sys)# snmpv2-ap
```

The SNMP v2 agent settings have been updated.

```
ruckus(config-sys)#
```

## no snmpv2-ap

To disable SNMP AP notification, use the following command:

```
no snmpv2-ap
```



```
ruckus(config-sys)# no snmpv2-ap
The SNMP v2 agent settings have been updated.
ruckus(config-sys)#
```

## SNMPv3 Commands

Use the following commands to configure SNMPv3 settings. To use these commands, you must first enter the `config-sys-snmpv3` context.

### snmpv3

To configure the SNMPv3 settings, use the following command:

```
snmpv3
```

Executing this command enters the `config-sys-snmpv3` context.

### Syntax Description

snmpv3	Configure the SNMPv3 settings
abort	Exits the <code>config-sys-snmpv3</code> context without saving changes.
end	Saves changes, and then exits the <code>config-sys-snmpv3</code> context.
exit	Saves changes, and then exits the <code>config-sys-snmpv3</code> context.
quit	Exits the <code>config-sys-snmpv3</code> context without saving changes.
ro-user <WORD>	Contains commands that can be executed from within the context.
ro-user <WORD> MD5 <WORD>	Contains commands that can be executed from within the context.
ro-user <WORD> MD5 <WORD> DES <WORD>	Sets the privacy phrase of DES for SNMPV3.
ro-user <WORD> MD5 <WORD> AES <WORD>	Sets the privacy phrase of AES for SNMPV3.

ro-user <WORD> MD5 <WORD> None	Sets the privacy to None for SNMPV3.
ro-user <WORD> SHA <WORD>	Contains commands that can be executed from within the context.
ro-user <WORD> SHA <WORD> DES <WORD>	Sets the privacy phrase of DES for SNMPV3.
ro-user <WORD> SHA <WORD> AES <WORD>	Sets the privacy phrase of AES for SNMPV3.
ro-user <WORD> SHA <WORD> None	Sets the privacy to None for SNMPV3.
rw-user <WORD>	Contains commands that can be executed from within the context.
rw-user <WORD> MD5 <WORD>	Contains commands that can be executed from within the context.
rw-user <WORD> MD5 <WORD> DES <WORD>	Sets the privacy phrase of DES for SNMPV3.
rw-user <WORD> MD5 <WORD> AES <WORD>	Sets the privacy phrase of AES for SNMPV3.
rw-user <WORD> MD5 <WORD> None	Sets the privacy to None for SNMPV3.
rw-user <WORD> SHA <WORD>	Contains commands that can be executed from within the context.
rw-user <WORD> SHA <WORD> DES <WORD>	Sets the privacy phrase of DES for SNMPV3.
rw-user <WORD> SHA <WORD> AES <WORD>	Sets the privacy phrase of AES for SNMPV3.
rw-user <WORD> SHA <WORD> None	Sets the privacy to None for SNMPV3.
show	Displays SNMPV3 agent and SNMP trap settings.

## Defaults

```
SNMPV3 Agent:
  Status= Disabled
  Ro:
    User=
    Authentication Type= MD5
    Authentication Pass Phrase=
    Privacy Type= DES
    Privacy Phrase=
  Rw:
    User=
    Authentication Type= MD5
    Authentication Pass Phrase=
    Privacy Type= DES
    Privacy Phrase=

SNMP Trap:
  Format= Version3
  Status= Disabled
```

## snmp-trap-format

To set the SNMP trap format to SNMPV2 or SNMPV3, use the following command:

```
snmp-trap-format [SNMPv2 | SNMPv3]
```

## Syntax Description

snmp-trap-format	Set the SNMP trap format
[SNMPv2   SNMPv3]	Set to either SNMPv2 or SNMPv3

## Defaults

SNMPv2

## Example

```
ruckus(config-sys) # snmp-trap-format SNMPV2
The SNMP trap settings have been updated.
```

## snmpv2-trap

To enable the SNMPv2 trap and set the IP address of the trap server, use the following command:

```
snmpv2-trap <NUMBER> <IP/IPv6-ADDR>
```

### Syntax Description

snmpv2-trap	Enable the SNMPv2 trap and set the trap server's IP address
<NUMBER>	Assign the trap receiver ID (1-4)
<IP/IPv6-ADDR>	Set the trap receiver IP address

### Defaults

None

### Example

```
ruckus(config-sys)# snmpv2-trap 1 192.168.10.22
```

The SNMP trap settings have been updated.

## snmpv3-trap

To enable and configure the SNMPv3 trap parameters, use the following command:

```
snmpv3-trap <user_name> <snmp_trap_server_ip> [MD5 | SHA]
<auth_pass_phrase> [DES <privacy_phrase>|AES <privacy_phrase>| None]
```

### Syntax Description

snmpv3-trap	Enable the SNMPv3 trap and configure the trap parameters
<user_name>	Trap user name
<snmp_trap_server_ip>	Trap server IP address
[MD5   SHA]	Authentication method
<auth_pass_phrase>	Authentication passphrase

---

```
[DES                                Privacy method and privacy phrase
<privacy_phrase>|AES
<privacy_phrase>| None]
```

---

**Defaults**

None

**Example**

```
ruckus(config-sys)#snmpv3-trap test1234 192.168.0.22 MD5 test1234
DES test4321
```

The command was executed successfully.

**no snmp-trap-ap**

To disable SNMP trap server configuration for AP, use the following command:

```
no snmp-trap-ap
```

**Example**

```
ruckus(config-sys)#no snmp-trap-ap
```

The SNMP AP trap settings have been updated.

**Syslog Settings Commands**

Use the `syslog` commands to configure the controller's syslog notification settings. To run these commands, you must first enter the `config-sys` context.

**no syslog**

To disable syslog notification, use the following command:

```
no syslog
```

**Syntax Description**


---

<code>no syslog</code>	Disable syslog notification
------------------------	-----------------------------

---

**Defaults**

Disabled.

### **Example**

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# no syslog
The command was executed successfully.
```

### **syslog**

To enable syslog notifications and enter the config-sys-syslog context, use the following command:

```
syslog
```

### **server**

To set the syslog server address, use the following command:

```
server <IP-ADDR>
```

### **Syntax Description**

<code>server</code>	Set the syslog server IP address.
<code>&lt;IPADDR&gt;</code>	Send syslog notifications to this IP address.

### **Defaults**

Disabled.

### **facility**

To set the facility name, use the following command:

```
facility <FACILITY NAME>
```

### **Syntax Description**

<code>facility</code>	Sets the syslog facility name (local0 - local7)
<code>&lt;FACILITY NAME&gt;</code>	

### **Defaults**

Disabled.

## priority

To set the syslog priority level, use the following command:

```
priority <PRIORITY LEVEL>
```

### ***Syntax Description***

---

priority	Sets the syslog priority level (emerg, alert, crit, err, warning, <PRIORITY LEVEL> notice, info, debug).
----------	--

---

### ***Defaults***

Disabled.

## ap-facility

To set the AP syslog facility name, use the following command:

```
ap-facility <FACILITY-NAME>
```

### ***Syntax Description***

---

ap-facility <FACILITY-NAME>	Sets the AP syslog facility name (local0 - local7).
-----------------------------	---

---

### ***Defaults***

Disabled.

## ap-priority

To set the AP syslog priority level, use the following command:

```
ap-priority <PRIORITY LEVEL>
```

### ***Syntax Description***

---

ap-priority	Sets the AP syslog priority level (emerg, alert, crit, err, <PRIORITY LEVEL> warning, notice, info, debug).
-------------	---

---

---

<IPADDR> Send syslog notifications to this IP address.

---

## **Defaults**

Disabled.

## **Example**

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# syslog
ruckus(config-sys-syslog)# server 192.168.3.10
The syslog settings have been updated.
ruckus(config-sys-syslog)# facility local0
The syslog settings have been updated.
ruckus(config-sys-syslog)# priority emerg
The syslog settings have been updated.
ruckus(config-sys-syslog)# ap-facility local0
The syslog settings have been updated.
ruckus(config-sys-syslog)# ap-priority emerg
The syslog settings have been updated.
ruckus(config-sys-syslog)# end
The syslog settings have been updated.
Your changes have been saved.
ruckus(config-sys)#
```

## **event-log-level**

To configure the event log level, use the following command:

```
event-log-level <EVENT LOG LEVEL>
```

## **Syntax Description**

---

event-log-level	Enter the syslog event log level <1-3> (1:Critical Events Only, 2:Warning and Critical Events, 3:Show More).
-----------------	--

---

## **Defaults**



## 2: Warning and Critical Events

### **Example**

```
ruckus# config
You have all rights in this mode.
ruckus(config)# sys
ruckus(config-sys)# syslog
ruckus(config-sys-syslog)# event-log-level 1
The syslog settings have been updated.
ruckus(config-sys-syslog)#
```

### **bypasscna**

Use the following command to bypass Apple Captive Network Assistance (CNA) on iPhones and OS X machines.

```
bypasscna <WLAN-TYPE>
```

### **Syntax Description**

bypasscna	Bypass Apple Captive Network Assistance (CNA) on iPhones and OS X machines
<WLAN-TYPE>	Enter the WLAN service type (web-auth, guestaccess, wispr)

### **Example**

```
ruckus(config-sys)# bypasscna web-auth
```

### **no bypasscna**

To disable the ignore Apple CNA feature, use the following command:

```
no bypasscna
```

### **Example**

```
ruckus(config-sys)# no bypasscna
```

## no syslog-ap

To disable external syslog server configuration for AP, use the following command:

```
no syslog-ap
```

### Example

```
ruckus(config-sys)#no syslog-ap
```

The AP syslog settings have been updated.

## Management Access Control List Commands

Use the following commands to create or configure management ACLs and enter the config-sys-mgmt-acl or config-sys-mgmt-acl-ipv6 contexts. These commands must be used from the config-sys context.

### mgmt-acl

To create or configure a management ACL, use the following command:

```
mgmt-acl <WORD>
```

Executing this command enters the config-mgmt-acl context.

### Syntax Description

mgmt-acl	Create or configure a management ACL
<WORD>	Create or configure this management ACL

### Defaults

None.

### Example

```
ruckus(config-sys)# mgmt-acl mac11
```

The management ACL 'mac11' has been created. To save the Management ACL, type 'end' or 'exit'.

```
ruckus(config-mgmt-acl)#
```

### no mgmt-acl

To delete a management ACL for IPv4, use the following command:

```
no mgmt-acl <WORD>
```

## mgmt-acl-ipv6

To create or configure an IPv6 management ACL, use the following command:

```
mgmt-acl-ipv6 <WORD>
```

Executing this command enters the `config-mgmt-acl-ipv6` context.

### *Syntax Description*

<code>mgmt-acl-ipv6</code>	Create or configure a management ACL
<code>&lt;WORD&gt;</code>	Create or configure this management ACL

### *Defaults*

None.

### *Example*

```
ruckus(config-sys)# mgmt-acl-ipv6 mac11
```

The management ACL 'mac11' has been created. To save the Management ACL, type 'end' or 'exit'.

```
ruckus(config-mgmt-acl-ipv6)#
```

## no mgmt-acl-ipv6

To delete a management ACL for IPv6, use the following command:

```
no mgmt-acl-ipv6 <WORD>
```

### **exit**

Saves changes, and then exits the `config-mgmt-acl` context.

### **end**

Saves changes, and then exits the `config-mgmt-acl` context.

### **quit**

Exits the `config-mgmt-acl` context without saving changes.

## abort

Exits the config-mgmt-acl context without saving changes.

## name

To set the management ACL name, use the following command:

```
name <WORD>
```

## restrict-type

To set the management ACL restriction type, use the following command:

```
restrict-type [single ip-addr <IP-ADDR> | range ip-range  
<IP-ADDR> <IP-ADDR> | subnet ip-subnet <IP-ADDR> <IP-  
SUBNET>]
```

## Syntax Description

restrict-type	Set the management ACL restriction type (single/range).
single ip-addr	Set management ACL restriction type to single.
range	Sets the management ACL restriction type to range.
ip-range	Sets the IP address range for management ACL. Use a space ( ) to separate addresses.
subnet ip-subnet	Sets the subnet for management ACL IP address. Use a space ( ) to separate IP address and Netmask (128.0.0.0 to 255.255.255.252).

## show

To display management ACL settings, use the show command.

## QoS Commands

Use the following commands to configure QoS settings on the controller. These commands must be executed from the config-sys context.

## no qos

To disable QoS on the controller, use the following command:

```
no qos
```

## Syntax Description

---

<code>no qos</code>	Disable QoS on the controller
---------------------	-------------------------------

---

## Defaults

None.

## Example

```
ruckus(config-sys) # no qos
Changes are saved!
System QoS function has been disabled.
```

## qos

To enable and configure Quality of Service settings on the controller, use the following command:

```
qos
```

Executing this command enters the `config-sys-qos` context. The following commands can be executed from within the `qos` context.

## Example

```
ruckus(config-sys) # qos
ruckus(config-sys-qos) #
```

## heuristics video inter-packet-gap

Use the following command to set the QoS heuristics video inter-packet gap minimum/maximum values:

```
heuristics video inter-packet-gap min <NUMBER> max
<NUMBER>
```

## heuristics video packet-length

Use the following command to set the heuristics video packet-length values:

```
heuristics video packet-length min <NUMBER> max <NUMBER>
```

## heuristics voice inter-packet-gap

Use the following command to set the heuristics voice inter-packet-gap values:

```
heuristics voice inter-packet-gap min <NUMBER> max  
<NUMBER>
```

## heuristics voice packet-length

Use the following command to set the heuristics voice packet-length values:

```
heuristics voice packet-length min <NUMBER> max <NUMBER>
```

## heuristics classification video packet-octet-count

Use the following command to set the heuristics classification video packet-octet-count value:

```
heuristics classification video packet-octet-count  
<NUMBER>
```

## heuristics classification voice packet-octet-count

Use the following command to set the heuristics classification voice packet-octet-count value:

```
heuristics classification voice packet-octet-count  
<NUMBER>
```

## heuristics no-classification video packet-octet-count

Use the following command to set the heuristics no-classification video packet-octet-count value

```
heuristics no-classification video packet-octet-count  
<NUMBER>
```

## heuristics no-classification voice packet-octet-count

Use the following command to set the heuristics no-classification voice packet-octet-count value

```
heuristics no-classification voice packet-octet-count  
<NUMBER>
```

## tos classification video

Use the following command to set the TOS classification video value:

```
tos classification video <WORD>
```

## tos classification voice

Use the following command to set the TOS classification voice value:

```
tos classification voice <WORD>
```

## tos classification data

Use the following command to set the TOS classification data value:

```
tos classification data <WORD>
```

## tos classification background

Use the following command to set the TOS classification background value:

```
tos classification background <WORD>
```

## show

Use the following command to display the system QoS settings:

```
show
```

## tunnel-mtu

To set the tunnel MTU, use the following command:

```
tunnel-mtu <NUMBER>
```

## *Syntax Description*

---

tunnel-mtu	Set the tunnel MTU
------------	--------------------

---

## *Defaults*

None.

## *Example*

```
ruckus(config-sys)# tunnel-mtu 1500  
The Tunnel MTU settings have been updated.  
ruckus(config-sys)#
```

## bonjour

To enable bonjour service, use the following command:

```
bonjour
```

### **Defaults**

Disabled.

### **Example**

```
ruckus(config-sys) # bonjour
```

The bonjour service settings have been updated.

```
ruckus(config-sys) #
```

### **no bonjour**

To disable bonjour service, use the following command:

```
no bonjour
```

### **telnetd**

To enable the telnet server, use the following command:

```
telnetd
```

### **Syntax Description**

---

telnetd	Enable the telnet server
---------	--------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(config-sys) # telnetd
```

The telnet server settings have been updated.

```
ruckus(config-sys) #
```

### **no telnetd**

To disable the telnet server, use the following command:

```
telnetd
```



## Syntax Description

---

<code>no telnetd</code>	Disable the telnet server
-------------------------	---------------------------

---

### Defaults

None.

### Example

```
ruckus(config-sys)# no telnetd
The telnet server settings have been updated.
ruckus(config-sys)#
```

## static-route

To create and configure static route settings, use the following command:

```
static-route <WORD>
```

## Syntax Description

---

<code>static-route</code>	Create and configure a static route
<code>name &lt;WORD&gt;</code>	Set the name of the static route
<code>subnet &lt;IP-SUBNET&gt;</code>	Set the subnet for the destination network. Use a slash (/ ) to separate IP address and subnet
<code>gateway &lt;GATEWAY-ADDR&gt;</code>	Set the gateway address
<code>show</code>	Show a list of all static routes

---

### Defaults

None.

### Example

```
ruckus(config-sys)# static-route routel
The static route 'routel' has been created. To save the static route, type 'end' or 'exit'.
```

```
ruckus(config-static-route)# subnet 192.168.11.1/24
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-static-route)# gateway 192.168.11.1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-static-route)# show
Static Route:
ID=
Name= route1
IP subnet= 192.168.11.1/24
IP gateway= 192.168.11.1

ruckus(config-static-route)#
```

## no static-route

To delete a static route, use the following command:

```
no static-route
```

## static-route-ipv6

To create and configure IPv6 static route settings, use the following command:

```
static-route-ipv6 <WORD>
```

## Syntax Description

static-route- ipv6	Create and configure a static route
name <WORD>	Set the name of the static route
prefix <IPv6-PREFIX>	Set the subnet for the destination network. Use a slash (/ ) to separate IP address and prefix length
gateway <GATEWAY- ADDR>	Set the gateway address
show	Show a list of all static routes

## Defaults

None.

### **Example**

```
ruckus(config-sys)# static-route routel
```

The static route 'routel' has been created. To save the static route, type 'end' or 'exit'.

```
ruckus(config-static-route)# subnet 192.168.11.1/24
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-static-route)# gateway 192.168.11.1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-static-route)# show
```

Static Route:

ID=

Name= routel

IP subnet= 192.168.11.1/24

IP gateway= 192.168.11.1

```
ruckus(config-static-route)#
```

### **no static-route-ipv6**

To delete an IPv6 static route, use the following command:

```
no static-route-ipv6 <WORD>
```

### **login-warning**

To configure the login warning message, use the following command:

```
login-warning
```

## Syntax Description

---

<code>login-warning</code>	Configure the login warning message.
<code>abort</code>	Exits the login-warning context without saving changes.
<code>end</code>	Saves changes, and then exits the login-warning context.
<code>exit</code>	Saves changes, and then exits the login-warning context.
<code>quit</code>	Exits the login-warning context without saving changes.
<code>content &lt;WORD&gt;</code>	Customize login warning content.

---

## Example

```
ruckus(config-sys)# login-warning
ruckus(config-sys-login-warning)# content "Warning, you are logging
into equipment belonging to ruckus, if you are not an authorized
user please logout immediately."
The login warning settings have been updated.
ruckus(config-sys-login-warning)# end
The login warning settings have been updated.
Your changes have been saved.
ruckus(config-sys)#
```

## no login-warning

To disable the login warning message, use the following command:

```
no login-warning
```

## show

Use the following command to display system configuration information:

```
show
```

## show support-entitle

To display the content of the entitlement file, use the following command:

```
show support-entitle
```

## Example

```
ruckus(config-sys)# show support-entitle
Serial Number: SN1150
```

```
Services purchased: 904  
Date to Start :Thu Oct 16 00:00:00 2014
```

```
Date to End: Wed Jan 14 23:59:00 2015
```

```
Number of APs: licensed  
Status: active  
Detailed: Support service activated  
ruckus(config-sys)#
```

## support-entitle

Use the following command to manually download entitlement file:

```
support-entitle
```

### **Example**

```
ruckus(config-sys)# support-entitle  
Your Support service has been successfully activated for this  
ZoneDirector. You may proceed with firmware upgrade.  
ruckus(config-sys)#
```

## session-stats-resv

To enable session statistics recording, use the following command:

```
session-stats-resv
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-sys)# session-stats-resv  
The session statistics function has been enabled.  
ruckus(config-sys)#
```

## no session-stats-resv

Use the following command to disable recording of session statistics:

```
no session-stats-resv
```

### **Example**

```
ruckus(config-sys)# no session-stats-resv
```

The session statistics function has been disabled.

```
ruckus(config-sys)#
```

### **session-limit-unauth-stats**

To enable recording of Layer 2 unauthorized session statistics, use the following command:

```
session-limit-unauth-stats
```

### **Defaults**

Enabled

### **Example**

```
ruckus(config-sys)# session-limit-unauth-stats
```

The limited unauthorized session statistics function has been enabled.

```
ruckus(config-sys)#
```

### **no session-limit-unauth-stats**

To disable recording of Layer 2 unauthorized session statistics, use the following command:

```
no session-limit-unauth-stats
```

### **no snmpv2**

To disable the SNMPv2 agent, use the following command:

```
no snmpv2
```

### **Syntax Description**

---

no snmpv2	Disables the SNMPv2 agent
-----------	---------------------------

---

### **Example**

```
ruckus(config-sys)# no snmpv2
```

The SNMP v2 agent settings have been updated.

### **no snmpv3**

To disable the SNMPv3 agent, use the following command:

```
no snmpv3
```

#### ***Syntax Description***

---

no snmpv3	Disables the SNMPv3 agent
-----------	---------------------------

---

#### ***Example***

```
ruckus(config-sys)# no snmpv3
```

The SNMP v3 agent settings have been updated.

### **no snmp-trap**

To disable the SNMP trap notifications, use the following command:

```
no snmp-trap <NUMBER>
```

#### ***Syntax Description***

---

no snmp-trap	Disables SNMP trap notification by index
--------------	--

---

#### ***Example***

```
ruckus(config-sys)# no snmp-trap 1
```

The SNMP trap settings have been updated.

### **no snmpv2-trap**

To disable the SNMP trap notifications, use the following command:

```
no snmp-trap <NUMBER>
```

#### ***Syntax Description***

---

no snmpv2-trap	Disables SNMP trap notification by index
----------------	--

---

#### ***Example***

```
ruckus(config-sys)# no snmpv2-trap 1
The SNMP trap settings have been updated.
```

## no snmpv3-trap

To disable the SNMPv3 trap notification, use the following command:

```
no snmpv3-trap <NUMBER>
```

### *Syntax Description*

---

no snmpv3-trap	Disables SNMP trap notification by index
----------------	--

---

### *Example*

```
ruckus(config-sys)# no snmpv3-trap 1
The SNMP trap settings have been updated.
```

## snmp-trap

To set the SNMP trap format, use the following command:

```
snmp-trap {trap server address}
```

### *Syntax Description*

---

snmp-trap	Enable SNMP trap notifications
{trap server address}	Set the trap server address to this IP address or host name

---

### *Example*

```
ruckus# config
ruckus(config)# system
ruckus(config-sys)# snmp-trap 192.168.0.3
```

## Management ACL Commands

Use the `mgmt-acl` commands to configure the management ACL settings. To run these commands, you must first enter the `config-mgmt-acl` context.



## abort

To exit the `config-mgmt-acl` context without saving changes, use the abort command.

```
abort
```

## end

To save changes, and then exit the `config-services` context, use the following command:

```
end
```

## exit

To save changes, and then exit the `config-services` context, use the following command:

```
exit
```

## quit

To exit the `config-mgmt-acl` context without saving changes, use the abort command.

```
quit
```

## name

To set the management ACL name, use the following command:

```
name <WORD>
```

## restrict-type single ip-addr

To set the management ACL restriction type to a single IP address, use the following command:

```
restrict-type single ip-addr <ip_address>
```

### ***Syntax Description***

---

<code>restrict-type single ip-addr</code>	Set the management ACL restriction type to a single IP address
<code>&lt;ip_address&gt;</code>	Set to this IP address only

---

**Example**

```
ruckus(config-mgmt-acl)# restrict-type single ip-addr
192.168.110.22
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

**restrict-type subnet ip-subnet**

To set the management ACL restriction type to certain subnets, use the following command:

```
restrict-type subnet ip-subnet <IP-SUBNET> <IP-SUBNET>
```

**Syntax Description**

restrict-type subnet ip-subnet	Set the management ACL restriction type to a single IP address
<IP-SUBNET>	Set to this subnet

**Example**

```
ruckus(config-mgmt-acl)#restrict-type subnet ip-subnet
172.30.110.26 255.255.254.0
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

**restrict-type range ip-range**

To set the management ACL restriction type to an IP address range, use the following command:

```
restrict-type range ip-range <ip_address> <ip_address>
```

**Syntax Description**

restrict-type range ip-range	Set the management ACL restriction type to a single IP address
<ip_address> <ip_address>	Set to this IP address range. The first <ip_address> is for the startui

**Example**

```
ruckus(config-mgmt-acl)#restrict-type range ip-range 172.30.110.28
172.30.110.39
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## show

To display the current management ACL settings, use the following command:

```
show
```

### *Syntax Description*

---

show	Display the current management ACL settings
------	---

---

### *Example*

```
ruckus(config-mgmt-acl)# show
Management ACL:
ID:
:
Name= macl2
Restriction Type= range
IP range= 172.30.110.28-172.30.110.39
```

## Configure UPNP Settings

Use the following commands to enable or disable Universal Plug and Play:

### **upnp**

```
upnp
```

### *Syntax Description*

---

upnp	Enable UPnP
------	-------------

---

### *Defaults*

Enabled.

### **Example**

```
ruckus(config)# upnp
UPnP Service is enabled
/bin/upnp enable
ruckus(config)#
```

### **no upnp**

```
no upnp
```

### **Syntax Description**

---

no upnp	Enable UPnP
---------	-------------

---

### **Defaults**

Enabled.

### **Example**

```
ruckus(config)# no upnp
UPnP Service is disabled
/bin/upnp disable
ruckus(config)#
```

## **Configure Zero-IT Settings**

To configure Zero-IT settings, use the following commands.

### **zero-it**

To configure Zero-IT settings, use the following command:

```
zero-it [local | name <WORD>]
```

### **zero-it-auth-server**

To configure Zero-IT settings, use the following command:

```
zero-it-auth-server [local | name <WORD>]
```

### Syntax Description

zero-it-auth-server	Set Zero-IT authentication server
local	Set the Zero-IT authentication server to local database
name	Set the Zero-IT authentication server to an external AAA server
<WORD>	Name of AAA server

### Defaults

None.

### Example

```
ruckus(config)# zero-it-auth-server name radius
The Authentication Server of Zero IT Activation has been updated.
ruckus(config)#
```

## Configure Dynamic PSK Expiration

The following section lists commands for configuring Dynamic Pre-Shared Keys.

### dynamic-psk-expiration

To set DPSK expiration, use the following command:

```
dynamic-psk-expiration <TIME>
```

### Syntax Description

dynamic-psk-expiration	Set DPSK expiration
<TIME>	Set DPSK expiration to this time limit (one-day, one-week, two-weeks, one-month, two-months, three-months, half-a-year, one-year, two-years)
unlimited	Set DPSKs to never expire

### Defaults

None.

### **Example**

```
ruckus(config)# dynamic-psk-expiration unlimited  
The Dynamic psk expiration value has been updated.  
ruckus(config)#
```

## **Configure WLAN Settings Commands**

Use the `config-wlan` commands to configure the WLAN settings, including the WLAN's description, SSID, and its security settings. To run these commands, you must first enter the `config-wlan` context.

### **wlan**

To create a WLAN or configure an existing WLAN, use the following command:

```
wlan <WORD/NAME>
```

Executing this command enters the `config-wlan` context.

### **Syntax Description**

---

wlan	Configure a WLAN
<WORD/NAME>	Name of the WLAN service

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan ruckus2  
The WLAN service 'ruckus2' has been created. To save the WLAN  
service, type 'end' or 'exit'.  
ruckus(config-wlan)#
```

### **abort**

Exits the `config-wlan` context without saving changes.

## **end**

Saves changes, and then exits the config-wlan context.

## **exit**

Saves changes, and then exits the config-wlan context.

## **quit**

Exits the config-wlan context without saving changes.

## **description**

To set the WLAN service description, use the following command:

```
description <WORD>
```

### ***Syntax Description***

---

description	Configure the WLAN description
<WORD>	Set the WLAN description this value

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# description ruckustestwlan2
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **called-station-id-type**

To set the called station ID type to, use the following command:

```
called-station-id-type [wlan-bssid | ap-mac]
```

### ***Syntax Description***

---

wlan-bssid	Set the called station ID type to 'BSSID:SSID'
------------	--

---

---

ap-mac	Set the called station ID type to 'APMAC:SSID'
--------	--

---

### **Defaults**

wlan-bssid

### **Example**

```
ruckus(config-wlan)# called-station-id-type wlan-bssid
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **ssid**

To set the WLAN service's SSID or network name, use the following command:

```
ssid <SSID>
```

### **Syntax Description**

---

ssid	Configure the WLAN service's SSID
<SSID>	Set the SSID to this value

---

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# ssid ruckus2
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **beacon-interval**

To set the beacon interval for mesh links, use the following command:

```
beacon-interval <NUMBER>
```



## Syntax Description

---

beacon-interval	Set the beacon interval for the WLAN
<NUMBER>	Enter the beacon interval (100~1000 TUs)

---

## Defaults

100

## Example

```
ruckus(config-wlan)# beacon-interval 100
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## mgmt-tx-rate

To set the transmit rate for management frames, use the following command:

```
mgmt-tx-rate <RATE>
```

## Syntax Description

---

mgmt-tx-rate	Set the max transmit rate for management frames
<RATE>	Set the transmit rate (in Mbps).

---

## Defaults

2

## Example

```
ruckus(config-wlan)# mgmt-tx-rate 2
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **name**

To set the name of the WLAN, use the following command:

```
name <NAME>
```

### ***Syntax Description***

---

name	Set the WLAN name
<NAME>	Set to this name

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# name ruckus2
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **type**

To configure the WLAN type, use the following command:

```
type [standard-usage | guest-access | hotspot <WORD> |  
hs20 <WORD> | autonomous]
```

### ***Syntax Description***

---

type	Set the WLAN type
standard-usage	Set the WLAN type to standard usage
guest-access	Set the WLAN type to guest access
hotspot <WORD>	Set the WLAN type to Hotspot using the hotspot service specified
hs20 <WORD>	Set the WLAN type to Hotspot 2.0 using the HS2.0 operator specified
autonomous	Set the WLAN type to Autonomous.

---

## **Defaults**

Standard usage

### **Example**

```
ruckus(config-wlan)# type standard-usage
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **type standard-usage**

To set the WLAN type to “Standard Usage”, use the following command:

```
type standard-usage  
type standard
```

### **type guest-access**

To set the WLAN type to “Guest Access”, use the following command:

```
type guest-access <WORD>
```

### **Example**

```
ruckus(config-wlan)# type guest-access guestpolicy1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **type hotspot**

To set the WLAN type to “Hotspot”, use the following command:

```
type hotspot
```

### **type hs20**

To set the WLAN type to “Hotspot 2.0”, use the following command:

```
type hs20
```

### **type autonomous**

To set the WLAN type to “Autonomous”, use the following command:

```
type autonomous
```

## **open none**

To set the authentication method to 'open' and encryption method to 'none', use the following command:

```
open none
```

### ***Syntax Description***

---

open	Set the authentication method to 'open'
none	Set the encryption method to 'none'

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config)# wlan wlan2
```

The WLAN service 'wlan2' has been created. To save the WLAN service, type 'end' or 'exit'.

```
ruckus(config-wlan)# open none
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)# end
```

The WLAN service 'wlan2' has been updated and saved.

Your changes have been saved.

```
ruckus(config)#
```

## **open wpa passphrase algorithm AES**

To set the authentication method to 'open', encryption method to 'WPA', and algorithm to 'AES', use the following command:

```
open wpa passphrase <PASSPHRASE> algorithm AES
```

### ***Syntax Description***

---

open	Set the authentication method to open
------	---------------------------------------

---

---

wpa	Set the encryption method to WPA
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm AES	Set the encryption algorithm to AES

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan wlan2
The WLAN service 'wlan2' has been created. To save the WLAN service,
type 'end' or 'exit'.
ruckus(config-wlan)# open wpa passphrase pass1234 algorithm AES
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-wlan)# end
The WLAN service 'wlan2' has been updated and saved.
Your changes have been saved.
ruckus(config)#
```

### **open wpa passphrase algorithm TKIP**

To set the authentication method to 'open', encryption method to 'WPA', and algorithm to 'TKIP', use the following command:

```
open wpa passphrase <PASSPHRASE> algorithm TKIP
```

### **Syntax Description**

---

open	Set the authentication method to open
wpa	Set the encryption method to WPA
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm TKIP	Set the encryption algorithm to TKIP

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan randy-wlansvc-01-open
```

The WLAN service 'randy-wlansvc-01-open' has been created. To save the WLAN service, type end or exit.

```
ruckus(config-wlan)# open wpa passphrase 12345678 algorithm TKIP
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **open wpa passphrase algorithm auto**

To set the authentication method to 'open', encryption method to 'WPA', and algorithm to 'Auto', use the following command:

```
open wpa passphrase <PASSPHRASE> algorithm auto
```

### **Syntax Description**

open	Set the authentication method to open
wpa	Set the encryption method to WPA
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm auto	Set the encryption algorithm to Auto

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan randy-wlansvc-01-open
```

The WLAN service 'randy-wlansvc-01-open' has been created. To save the WLAN service, type end or exit.

```
ruckus(config-wlan)# open wpa passphrase 12345678 algorithm auto
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## open wpa2 passphrase algorithm AES

To set the authentication method to 'open', encryption method to 'WPA2', and algorithm to 'AES', use the following command:

```
open wpa2 passphrase <PASSPHRASE> algorithm AES
```

### Syntax Description

open	Set the authentication method to open
wpa2	Set the encryption method to WPA2
passphrase <PASSPHRASE>	Set the WPA2 passphrase to <PASSPHRASE>
algorithm AES	Set the encryption algorithm to AES

### Defaults

None.

### Example

```
ruckus(config)# wlan randy-wlansvc-01-open
```

The WLAN service 'randy-wlansvc-01-open' has been created. To save the WLAN service, type end or exit.

```
ruckus(config-wlan)# open wpa2 passphrase 12345678 algorithm AES
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## open wpa2 passphrase algorithm TKIP

To set the authentication method to 'open', encryption method to 'WPA2', and algorithm to 'TKIP', use the following command:

```
open wpa2 passphrase <PASSPHRASE> algorithm TKIP
```

### Syntax Description

open	Set the authentication method to open
wpa2	Set the encryption method to WPA2

passphrase <PASSPHRASE>	Set the WPA2 passphrase to <PASSPHRASE>
algorithm TKIP	Set the encryption algorithm to TKIP

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan randy-wlansvc-01-open
```

The WLAN service 'randy-wlansvc-01-open' has been created. To save the WLAN service, type end or exit.

```
ruckus(config-wlan)# open wpa2 passphrase 12345678 algorithm TKIP
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **open wpa2 passphrase algorithm auto**

To set the authentication method to 'open', encryption method to 'WPA2', and algorithm to 'Auto', use the following command:

```
open wpa2 passphrase <PASSPHRASE> algorithm auto
```

### **Syntax Description**

open	Set the authentication method to open
wpa2	Set the encryption method to WPA2
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm auto	Set the encryption algorithm to Auto

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan randy-wlansvc-01-open
```



The WLAN service 'randy-wlansvc-01-open' has been created. To save the WLAN service, type end or exit.

```
ruckus(config-wlan)# open wpa2 passphrase 12345678 algorithm auto  
The command was executed successfully.  
ruckus(config-wlan)#
```

## **open wpa-mixed passphrase algorithm auto**

To set the authentication method to 'open', encryption method to 'WPA mixed', and algorithm to 'Auto', use the following command:

```
open wpa-mixed passphrase <PASSPHRASE> algorithm [AES |  
TKIP | auto]
```

### ***Syntax Description***

open	Set the authentication method to open
wpa-mixed	Set the encryption method to WPA-mixed
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm AES	Set the encryption algorithm to AES
algorithm TKIP	Set the encryption algorithm to TKIP
algorithm auto	Set the encryption algorithm to Auto

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# open wpa-mixed passphrase pass1234 algorithm  
auto  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-wlan)#
```

## open wep-64 key {KEY} key-id {KEY-ID}

To set the authentication method to 'open', encryption method to 'WEP-64', key index, and WEP key, use the following command:

```
open wep-64 key {key} key-id {key ID}
```

### Syntax Description

open	Set the authentication method to open
wep-64	Set the encryption method to WEP 64-bit
key {key}	Set the WEP key to {key}
key-id {key ID}	Set the WEP key ID to {key ID}

### Defaults

None.

### Example

```
ruckus(config)# wlan wlan2
```

The WLAN service 'wlan2' has been created. To save the WLAN service, type 'end' or 'exit'.

```
ruckus(config-wlan)# open wep-64 key 1234567890 key-id 1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## open wep-128 key key-id

To set the authentication method to 'open', encryption method to 'WEP-128', key index, and WEP key, use the following command:

```
open wep-128 key {key} key-id {key ID}
```

### Syntax Description

open	Set the authentication method to open
wep-128	Set the encryption method to WEP 128-bit

---

key {key}	Set the WEP key to {key}
key-id {key ID}	Set the WEP key ID to {key ID}

---

### **Defaults**

None.

### **Example**

```
ruckus(config)# wlan wlan2
The WLAN service 'wlan2' has been loaded. To save the WLAN service,
type 'end' or 'exit'.
ruckus(config-wlan)# open wep-128 key 11111111111111111111111111111111
key-id 1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-wlan)#
```

### **mac none auth-server**

To set the authentication method to 'MAC Address' and encryption method to 'none', use the following command:

```
mac none auth-server <WORD>
```

### **Syntax Description**

---

mac	Set the authentication method to 'MAC Address'
none	Set the encryption method to 'none'
auth-server <WORD>	Set the authorization server address to <WORD>

---

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# mac none auth-server Ruckus-Auth-01
The command was executed successfully.
```

```
ruckus(config-wlan)#
```

## **mac wpa passphrase algorithm AES auth-server**

To set the authentication method to 'MAC Address', encryption method to 'WPA', and algorithm to 'AES', use the following command:

```
mac wpa passphrase <PASSPHRASE> algorithm AES auth-server  
<WORD>
```

### ***Syntax Description***

mac	Set the authentication method to 'MAC Address'
wpa	Set the encryption method to 'WPA'
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm AES	Set the encryption algorithm to 'AES'
auth-server <WORD>	Set the authorization server address to <WORD>

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# mac wpa passphrase 12345678 algorithm AES  
auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **mac wpa passphrase algorithm TKIP auth-server**

To set the authentication method to 'MAC Address', encryption method to 'WPA', and algorithm to 'TKIP', use the following command:

```
mac wpa passphrase <PASSPHRASE> algorithm TKIP auth-server  
<WORD>
```

## Syntax Description

mac wpa	Set the authentication method to 'MAC Address' and encryption method to 'WPA'
passphrase <PASSPHRASE>	Set the WPA passphrase to <PASSPHRASE>
algorithm TKIP	Set the encryption algorithm to 'TKIP'
auth-server <WORD>	Set the authorization server address to <WORD>

## Defaults

None.

## Example

```
ruckus(config-wlan)# mac wpa passphrase 12345678 algorithm TKIP  
auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## mac wpa2 passphrase algorithm AES auth-server

To set the authentication method to 'MAC Address', encryption method to 'WPA2', and algorithm to 'AES', use the following command:

```
mac wpa2 passphrase <PASSPHRASE> algorithm AES auth-server  
<WORD>
```

## Syntax Description

mac wpa2	Set the authentication method to 'MAC Address' and encryption method to 'WPA2'
passphrase <PASSPHRASE>	Set the WPA2 passphrase to <PASSPHRASE>
algorithm AES	Set the encryption algorithm to 'AES'
auth-server <WORD>	Set the authorization server address to <WORD>

## Defaults

None.

### **Example**

```
ruckus(config-wlan)# mac wpa2 passphrase 12345678 algorithm AES  
auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **mac wpa2 passphrase algorithm TKIP auth-server**

To set the authentication method to 'MAC Address', encryption method to 'WPA2', and algorithm to 'TKIP', use the following command:

```
mac wpa2 passphrase <PASSPHRASE> algorithm TKIP auth-server  
<WORD>
```

### **Syntax Description**

mac wpa2	Set the authentication method to 'MAC Address' and encryption method to 'WPA2'
passphrase <PASSPHRASE>	Set the WPA2 passphrase to <PASSPHRASE>
algorithm TKIP	Set the encryption algorithm to 'TKIP'
auth-server <WORD>	Set the authorization server address to <WORD>

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# mac wpa2 passphrase 12345678 algorithm TKIP  
auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **mac wpa-mixed passphrase algorithm AES auth-server**

To set the authentication method to 'MAC Address', encryption method to WPA-Mixed, and algorithm to AES, use the following command:

```
mac wpa-mixed passphrase <PASSPHRASE> algorithm AES auth-
server <WORD>
```

### ***Syntax Description***

mac wpa-mixed	Set the authentication method to 'MAC Address' and encryption method to 'WPA-Mixed'
passphrase <PASSPHRASE>	Set the WPA2 passphrase to <PASSPHRASE>
algorithm AES	Set the encryption algorithm to 'AES'
auth-server <WORD>	Set the authorization server to this auth server

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# mac wpa-mixed passphrase pass1234 algorithm
AES auth-server radius
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **mac wpa-mixed passphrase algorithm TKIP auth-server**

To set the authentication method to 'MAC Address', encryption method to 'WPA-Mixed', algorithm to TKIP, use the following command:

```
mac wpa-mixed passphrase <PASSPHRASE> algorithm TKIP auth-
server <WORD>
```

### ***Syntax Description***

mac wpa-mixed	Set the authentication method to 'MAC Address' and encryption method to 'WPA-Mixed'
---------------	---

---

passphrase <PASSPHRASE>	Set the WPA2 passphrase to <PASSPHRASE>
algorithm TKIP	Set the encryption algorithm to 'TKIP'
auth-server <WORD>	Set the authorization server to this auth server

---

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# mac wpa-mixed passphrase pass1234 algorithm  
TKIP auth-server radius  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-wlan)#
```

### **mac wep-64 key key-id auth-server**

To set the authentication method to 'MAC Address', encryption method to 'WEP-64', key index, and WEP key, use the following command:

```
mac wep-64 key {KEY} key-id {KEY-ID} auth-server <WORD>
```

### **Syntax Description**

---

mac	Set the authentication method to MAC address
wep-64	Set the encryption method to WEP 64-bit
key {KEY}	Set the WEP key to {KEY}
key-id {KEY-ID}	Set the WEP key ID to {KEY-ID}
auth-server <WORD>	Set the authorization server address to <WORD>

---

### **Defaults**

None.

### **Example**



```
ruckus(config-wlan)# mac wep-64 key 15791BD8F2 key-id 2 auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **mac wep-128 key key-id auth-server**

To set the authentication method to 'MAC Address', encryption method to 'WEP-128', key index, and WEP key, use the following command:

```
mac wep-128 key {KEY} key-id {KEY-ID} auth-server <WORD>
```

### ***Syntax Description***

mac	Set the authentication method to MAC address
wep-128	Set the encryption method to WEP 128-bit
key {KEY}	Set the WEP key to {key}
key-id {KEY-ID}	Set the WEP key ID to {key ID}
auth-server <WORD>	Set the authorization server address to <WORD>

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# mac wep-128 key 15715791BD8F212345691BD8F2 key-id 2 auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **dot1x eap-type EAP-SIM auth-server**

To set the authentication method to 'EAP-SIM', use the following command:

```
dot1x eap-type EAP-SIM auth-server[local | name <WORD>]
```

### ***Syntax Description***

dot1x	Set the authentication method to '802.11x'
-------	--

eap-type	Set the EAP type
EAP-SIM	Set the authentication method to EAP-SIM
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# dot1x eap-type EAP-SIM auth-server local
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **dot1x eap-type PEAP auth-server**

To set the authentication method to 'PEAP', use the following command:

```
dot1x eap-type PEAP auth-server [local | name <WORD>]
```

### **Syntax Description**

dot1x	Set the authentication method to '802.11x'
eap-type	Set the EAP type
PEAP	Set the authentication method to PEAP
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# dot1x eap-type PEAP auth-server local
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## **dot1x wpa algorithm AES auth-server**

To set the authentication method to '802.1x EAP', encryption method to 'WPA', and algorithm to 'AES', use the following command:

```
dot1x wpa algorithm AES auth-server [local | name <WORD>]
```

### ***Syntax Description***

dot1x	Set the authentication method to '802.11x'
wpa	Set the encryption method to WPA
algorithm AES	Set the algorithm to AES
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# dot1x wpa algorithm AES auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **dot1x wpa algorithm TKIP auth-server**

To set the authentication method to '802.1x EAP', encryption method to 'WPA', and algorithm to 'TKIP', use the following command:

```
dot1x wpa algorithm TKIP auth-server <WORD>
```

## ***Syntax Description***

dot1x	Set the authentication method to '802.11x'
wpa	Set the encryption method to WPA
algorithm TKIP	Set the algorithm to TKIP
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

## ***Defaults***

None.

## ***Example***

```
ruckus(config-wlan)# dot1x wpa algorithm TKIP auth-server Ruckus-Auth-01
```

The command was executed successfully.

## **dot1x wpa algorithm auto auth-server**

To set the authentication method to '802.1x EAP', encryption method to 'WPA', and algorithm to 'Auto', use the following command:

```
dot1x wpa algorithm auto auth-server [local | name <WORD>]
```

## ***Syntax Description***

dot1x	Set the authentication method to '802.11x'
wpa	Set the encryption method to WPA
algorithm auto	Set the algorithm to Auto
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

## Defaults

None.

## Example

```
ruckus(config-wlan)# dot1x wpa algorithm auto auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## dot1x wpa2 algorithm AES auth-server

To set the authentication method to '802.1x EAP', encryption method to 'WPA2', and algorithm to 'AES', use the following command:

```
dot1x wpa2 algorithm AES auth-server [local | name <WORD>]
```

## Syntax Description

dot1x	Set the authentication method to '802.1x'
wpa2	Set the encryption method to WPA2
algorithm AES	Set the algorithm to AES
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

## Defaults

None.

## Example

```
ruckus(config-wlan)# dot1x wpa2 algorithm AES auth-server Ruckus-RADIUS
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## dot1x wpa2 algorithm TKIP auth-server

To set the authentication method to '802.1x EAP', encryption method to 'WPA2', and algorithm to 'TKIP', use the following command:

```
dot1x wpa2 algorithm TKIP auth-server [local | name <WORD>]
```

### Syntax Description

dot1x	Set the authentication method to '802.11x'
wpa2	Set the encryption method to WPA2
algorithm TKIP	Set the algorithm to TKIP
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### Defaults

None.

### Example

```
ruckus(config-wlan)# dot1x authentication encryption wpa2 algorithm  
TKIP auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## dot1x wpa2 algorithm auto auth-server

To set the authentication method to '802.1x EAP', encryption method to 'WPA2', and algorithm to 'Auto', use the following command:

```
dot1x wpa2 algorithm auto auth-server [local | name <WORD>]
```

### Syntax Description

dot1x	Set the authentication method to '802.11x'
wpa2	Set the encryption method to WPA2
algorithm auto	Set the algorithm to auto

auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# dot1x wpa2 algorithm auto auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **dot1x wpa-mixed algorithm AES auth-server**

To set the authentication method to 802.1x EAP, encryption method to WPA-Mixed, and encryption method to AES, use the following command:

```
dot1x wpa-mixed algorithm AES auth-server [local | name <WORD>]
```

### **Syntax Description**

dot1x	Set the authentication method to '802.11x'
wpa-mixed	Set the encryption method to WPA-Mixed
algorithm AES	Set the algorithm to AES
auth-server	Set authentication server
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# dot1x wpa-mixed algorithm AES auth-server local
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **dot1x wpa-mixed algorithm TKIP auth-server**

To set the authentication method to 802.1x EAP, encryption method to WPA-Mixed, and encryption method to TKIP, use the following command:

```
dot1x wpa-mixed algorithm TKIP auth-server [local | name <WORD>]
```

### **Syntax Description**

dot1x	Set the authentication method to '802.11x'
wpa-mixed	Set the encryption method to WPA-Mixed
algorithm TKIP	Set the algorithm to TKIP
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# dot1x wpa-mixed algorithm AES auth-server local
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

### **dot1x wpa-mixed algorithm auto auth-server**

To set the authentication method to 802.1x EAP, encryption method to WPA-Mixed, and encryption method to Auto, use the following command:



```
dot1x wpa-mixed algorithm auto auth-server [local | name  
<WORD>]
```

### ***Syntax Description***

dot1x	Set the authentication method to '802.11x'
wpa-mixed	Set the encryption method to WPA-Mixed
algorithm auto	Set the algorithm to Auto
local	Set the authentication server to 'local database'
name	Set the auth server
<WORD>	Name of the auth server

### ***Defaults***

None.

### ***Example***

```
ruckus(config-wlan)# dot1x wpa-mixed algorithm AES auth-server  
local  
The command was executed successfully.  
ruckus(config-wlan)#
```

### **dot1x authentication encryption wep-64 auth-server**

To set the authentication method to '802.1x EAP', encryption method to 'WEP-64', key index, and WEP key, use the following command:

```
dot1x authentication encryption wep-64 auth-server {auth  
server}
```

### ***Syntax Description***

dot1x authentication	Set the authentication method to '802.11x'
encryption wep-64	Set the encryption method to WEP 64-bit
auth-server {auth server}	Set the auth server to {auth server}

## **Defaults**

None.

## **Example**

```
ruckus(config-wlan)# dot1x authentication encryption wep-64 auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **dot1x wep-128 auth-server**

To set the authentication method to '802.1x EAP', encryption method to 'WEP-128', key index, and WEP key, use the following command:

```
dot1x wep-128 auth-server [local|name <WORD>]
```

## **Syntax Description**

---

dot1x	Set the authentication method to '802.11x'
wep-128	Set the encryption method to WEP 128-bit
auth-server [local name<WORD>]	Set the auth server to local or to the named server

---

## **Defaults**

None.

## **Example**

```
ruckus(config-wlan)# dot1x authentication encryption wep-128 auth-server Ruckus-Auth-01
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **dot1x none**

To set the encryption as none and authentication server to 'Local Database' or the named server, use the following command:

```
dot1x none auth-server [local|name<WORD>]
```

## Syntax Description

---

dot1x none	Set the authentication method to '802.1x' and encryption to none
------------	--

---

auth-server [local name<WORD>]	Set the auth server to local or to the named server
-----------------------------------	---

---

## Defaults

None.

## Example

```
ruckus(config-wlan)# dot1x none auth-server Ruckus-Auth-01  
The command was executed successfully.  
ruckus(config-wlan)#
```

## dot1x-mac none

To set the encryption as none and authentication method to 802.1x-MAC, use the following command:

```
dot1x-mac none auth-server name <WORD>
```

## Syntax Description

---

dot1x-mac none	Set the authentication method to '802.1x-MAC' and encryption to none
----------------	--

---

auth-server name<WORD>	Set the auth server to the named server
---------------------------	---

---

## Defaults

None.

## Example

```
ruckus(config-wlan)# dot1x-mac none auth-server Ruckus-Auth-01  
The command was executed successfully.  
ruckus(config-wlan)#
```

## **bgscan**

To enable background scanning on the WLAN, use the following command:

```
bgscan
```

### ***Example***

```
ruckus(config-wlan)# bgscan
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no bgscan**

To disable background scanning on the WLAN, use the following command:

```
no bgscan
```

### ***Example***

```
ruckus(config-wlan)# no bgscan
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **ft-roaming**

To enable FT Roaming, use the following command:

```
ft-roaming
```

### ***Example***

```
ruckus(config-wlan)# ft-roaming
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no ft-roaming**

To disable FT Roaming, use the following command:

```
no ft-roaming
```

## rrm-neigh-report

To enable 802.11k Neighbor-list report, use the following command:

```
rrm-neigh-report
```

### **Example**

```
ruckus(config-wlan)# rrm-neigh-report
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## no rrm-neigh-report

To isable 802.11k Neighbor-list report, use the following command:

```
no rrm-neigh-report
```

## https-redirection

To enable HTTPS redirection, use the following command:

```
https-redirection
```

## no https-redirection

To disable HTTPS redirection, use the following command:

```
no https-redirection
```

## client-isolation

To enable client isolation (per-AP or across APs, use the following command:

```
client-isolation [isolation-on-ap|isolation-on-subnet]  
[enable|disable]
```

### **Syntax Description**

client-isolation	Enable client isolation for this WLAN.
isolation-on-ap	Enable client isolation per AP.
isolation-on-subnet	Enable client isolation across APs.

### **Example**

```
ruckus(config-wlan)# client-isolation isolation-on-ap enable
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **whitelist**

To apply a client isolation whitelist to this WLAN, use the following command:

```
whitelist name <WORD>
```

## **no whitelist**

To disable the whitelist for this WLAN, use the following command:

```
no whitelist
```

## **load-balancing**

To enable load balancing for this WLAN, use the following command:

```
load-balancing
```

## **Defaults**

Disabled

## **Example**

```
ruckus(config-wlan)# load-balancing
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no load-balancing**

To disable load balancing for this WLAN, use the following command:

```
no load-balancing
```

## **Example**

```
ruckus(config-wlan)# no load-balancing
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **band-balancing**

To enable band balancing for this WLAN, use the following command:

```
band-balancing
```

### **Defaults**

Enabled.

### **Example**

```
ruckus(config-wlan)# band-balancing
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no band-balancing**

To disable band balancing for this WLAN, use the following command:

```
no band-balancing
```

## **send-eap-failure**

To enable send EAP failure messages, use the following command:

```
send-eap-failure
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-wlan)# send-eap-failure
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no send-eap-failure**

To disable send EAP failure messages, use the following command:

```
no send-eap-failure
```

### ***Example***

```
ruckus(config-wlan)# no send-eap-failure
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **pap-authenticator**

To enable RADIUS message authenticator in PAP requests, use the following command:

```
pap-authenticator
```

### ***Example***

```
ruckus(config-wlan)# pap-authenticator
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **no pap-authenticator**

To disable RADIUS message authenticator in PAP requests, use the following command:

```
no pap-authenticator
```

### ***Example***

```
ruckus(config-wlan)# no pap-authenticator
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **nasid-type**

To set the NAS ID type, use the following command:

```
nasid-type [wlan-bssid|mac-addr|user-define <WORD>]
```



## ***Syntax Description***

<code>nasid-type</code>	Set the NAS ID type
<code>wlan-bssid</code>	Set NAS ID type WLAN-BSSID (default)
<code>mac-addr</code>	Set NAS ID type to Controller MAC Address
<code>user-define &lt;WORD&gt;</code>	Set NAD ID type to a user-defined string

## ***Defaults***

WLAN-BSSID

## ***Example***

```
ruckus(config-wlan)# nasid-type wlan-bssid
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **priority low**

To set the WLAN priority to low, use the following command:

```
priority low
```

## **priority high**

To set the WLAN priority to high, use the following command:

```
priority high
```

## **web-auth**

To enable Web authentication, use the following command:

```
web-auth [local | name <WORD>]
```

## ***Syntax Description***

<code>web-auth</code>	Enable Web authentication
<code>local</code>	Use local database as auth server
<code>name</code>	Specify an external auth server
<code>&lt;WORD&gt;</code>	The AAA server to use for Web authentication

## **Defaults**

None

## **Example**

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# web-auth Ruckus-RADIUS
```

The command was executed successfully.

```
ruckus(config-wlan)#
```

## **no web-auth**

To disable Web authentication, use the following command:

```
no web-auth
```

## **Syntax Description**

---

no web-auth	Disable Web authentication
-------------	----------------------------

---

## **Defaults**

None.

## **Example**

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# no web-auth
```

The command was executed successfully.

## **grace-period**

To enable and set a maximum time (in minutes) for which users must re-authenticate after disconnecting, use the following command:

```
grace-period <NUMBER>
```

## ***Syntax Description***

---

<code>grace-period</code>	Enables and Sets a maximum time (in minutes) for which users must re-authenticate after disconnecting.
---------------------------	--

---

### ***Defaults***

Disabled.

### ***Example***

```
ruckus(config-wlan)# grace-period 20
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## **no grace-period**

To disable the grace period, use the following command:

```
no grace-period <NUMBER>
```

## ***Syntax Description***

---

<code>no grace-period</code>	Disables the grace period timeout.
------------------------------	------------------------------------

---

### ***Defaults***

Disabled.

### ***Example***

```
ruckus(config-wlan)# no grace-period
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## **acct-server**

To set the accounting server, use the following command:

```
acct-server <WORD>
```

## Syntax Description

---

acct-server	Configure the AAA server
<WORD>	Set the AAA server to this address

---

## Defaults

None.

## Example

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# acct-server Ruckus-Acct-01
The command was executed successfully.
```

## acct-server interim-update

To configure the interim update frequency (in minutes) of the AAA server, use the following command:

```
acct-server <WORD> interim-update <NUMBER>
```

## Syntax Description

---

acct-server	Configure the interim update frequency of the AAA server
interim-update{minutes}	Set the update frequency to this value (in minutes)

---

## Defaults

5 (minutes)

## Example

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# acct-server Ruckus-Acct-01 interim-update 5
The command was executed successfully.
```

## no acct-server

To disable the AAA server, use the following command:

```
no acct-server
```

### *Syntax Description*

---

no acct-server	Disable AAA server authentication
----------------	-----------------------------------

---

### *Defaults*

None.

### *Example*

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# no acct-server
The command was executed successfully.
```

## inactivity-timeout

To enable and set the inactivity timeout to the specified number in minutes, use the following command:

```
inactivity-timeout <NUMBER>
```

### *Syntax Description*

---

inactivity-timeout	Enable and set the inactivity timeout
<NUMBER>	Set the inactivity timeout in minutes

---

### *Defaults*

5

### *Example*

```
ruckus(config-wlan)# inactivity-timeout 15
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.  
ruckus(config-wlan)#

## **vlan**

To set the VLAN ID for the WLAN, use the following command:

```
vlan <NUMBER>
```

### ***Syntax Description***

---

vlan	Enable VLAN
<NUMBER>	Set the VLAN ID to this value

---

### ***Defaults***

1

### ***Example***

```
ruckus(config-wlan)# vlan 123  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-wlan)#
```

## **dynamic-vlan**

To enable dynamic VLAN, use the following command:

```
dynamic-vlan
```

### ***Syntax Description***

---

dynamic-vlan	Enable dynamic VLAN
--------------	---------------------

---

### ***Notes***

Dynamic VLAN can be enabled or disabled in the following two conditions: 1) The authentication method is '802.1X/EAP' or 'MAC Address', Encryption method is WPA, WPA2, WPA mixed, or none. 2) Authentication method is 'Open', Encryption method is WPA, WPA2 (Algorithm may not be Auto), enable Zero-IT Activation, enable Dynamic PSK.

### ***Example***

```
ruckus(config-wlan)# dynamic-vlan
```

The command was executed successfully. To save the changes, type 'end' or 'exit'

### **no dynamic-vlan**

To disable dynamic VLAN, use the following command:

```
no dynamic-vlan
```

### ***Syntax Description***

---

no dynamic-vlan	Disable dynamic VLAN
-----------------	----------------------

---

### ***Defaults***

Disabled.

### ***Example***

```
ruckus(config-wlan)# no dynamic-vlan
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **mcast-filter**

To enable multicast filter for the WLAN, use the following command:

```
mcast-filter
```

### **no mcast-filter**

To disable multicast filter for the WLAN, use the following command:

```
no mcast-filter
```

## hide-ssid

To hide an SSID from wireless users, use the following command. Wireless users who know the SSID will still be able to connect to the WLAN service.

```
hide-ssid
```

### ***Syntax Description***

---

hide-ssid	Hide SSID from wireless users
-----------	-------------------------------

---

### ***Defaults***

Disabled

### ***Example***

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# hide-ssid
The command was executed successfully.
```

## no hide-ssid

To unhide or broadcast an SSID to wireless users, use the following command:

```
no hide-ssid
```

### ***Syntax Description***

---

no hide-ssid	Broadcast SSID to wireless users
--------------	----------------------------------

---

### ***Defaults***

Disabled

### ***Example***

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# no hide-ssid
The command was executed successfully
```



## **ofdm-only**

To enable support of OFDM rates only, use the following command:

```
ofdm-only
```

## **no ofdm-only**

To disable OFDM only rates, use the following command:

```
no ofdm-only
```

## **admission-control**

To enable Call Admission Control, use the following command:

```
admission-control
```

## **no admission-control**

To disable Call Admissino Control, use the following command:

```
no admission-control
```

## **bss-minrate**

To set the minimum BSS transmission rate of the WLAN (in Mbps), use the following command:

```
bss-minrate <NUMBER>
```

## ***Syntax Description***

bss-minrate	Set the minimum BSS transmission rate in Mbps.
<NUMBER>	Minimum BSS transmission rate

## ***Defaults***

None.

## ***Example***

```
ruckus(config-wlan)# bss-minrate 2
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## no bss-minrate

To disable the minimum BSS transmission rate for the WLAN, use the following command:

```
no bss-minrate
```

## tunnel-mode

To enable tunnel mode, use the following command:

```
tunnel-mode
```

### *Syntax Description*

---

tunnel-mode	Enable tunnel mode
-------------	--------------------

---

### *Defaults*

Disabled.

### *Example*

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan)# tunnel-mode
The command was executed successfully.
```

## no tunnel-mode

To disable the tunnel mode, use the following command:

```
no tunnel-mode
```

### *Syntax Description*

---

no tunnel-mode	Disable the tunnel mode
----------------	-------------------------

---

### *Defaults*

Disabled.

### *Example*

```
ruckus# config
ruckus(config)# wlan wlan-123
ruckus(config-wlan-wlan-123)# no tunnel-mode
The command was executed successfully.
```

## **dhcp-relay**

To set the DHCP relay server to the specified address (tunneled WLANs only), use the following command:

```
dhcp-relay <WORD>
```

## **no dhcp-relay**

To disable DHCP relay, use the following command:

```
no dhcp-relay
```

## **smart-roam**

To enable and set SmartRoam with the specified roam factor (1-10), use the following command:

```
smart-roam <NUMBER/EMPTY>
```

## **no smart-roam**

To disable the SmartRoam feature, use the following command:

```
no smart-roam
```

## **force-dhcp**

To enable the Force DHCP option, use the following command:

```
force-dhcp
```

## **Defaults**

Disabled

## **Example**

```
ruckus(config-wlan)# force-dhcp
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-wlan)#
```

## **force-dhcp-timeout**

To disconnect the client if it does not obtain valid IP address within the specified timeout period (in seconds), use the following command:

```
force-dhcp-timeout <NUMBER>
```

### **Defaults**

10 seconds

### **Example**

```
ruckus(config-wlan)# force-dhcp-timeout 10
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no force-dhcp**

To disable the Force DHCP option, use the following command:

```
no force-dhcp
```

## **Configuring DHCP Option 82 Sub-Option Settings**

Use the following commands to enable DHCP Option 82 and configure sub-option settings for a WLAN.

### **option82**

To enable DHCP option 82 and enter the `config-wlan-option82` context, use the following command:

```
option82
```

### **Defaults**

Disabled

## ***Syntax Description***

subopt1	Enables and sets the DHCP option 82 sub-option1.
subopt1 disable	Disables the DHCP option 82 sub-option1.
subopt1 rks-circuitid	sets the DHCP option 82 sub-option1 is RKS_CircuitID.
subopt1 ap-mac-hex	sets the DHCP option 82 sub-option1 is AP-MAC.
subopt1 ap-mac-hex-ssid	sets the DHCP option 82 sub-option1 is AP-MAC and ESSID.
subopt2	Enables and sets the DHCP option 82 sub-option2.
subopt2 disable	Disables the DHCP option 82 sub-option2.
subopt2 client-mac-hex	sets the DHCP option 82 sub-option2 is Client-Mac.
subopt2 client-mac-hex-ssid	sets the DHCP option 82 sub-option2 is Client-Mac and Essid.
subopt2 ap-mac-hex	sets the DHCP option 82 sub-option2 is AP-MAC.
subopt2 ap-mac-hex-ssid	sets the DHCP option 82 sub-option2 is AP-MAC and ESSID.
subopt2 cuid	Sets the DHCP option 82 sub-option2 is CUID.
subopt150	Enables and sets the DHCP option 82 sub-option150.
subopt150 disable	Disables the DHCP option 82 sub-option150.
subopt150 vlan-id	sets the DHCP option 82 sub-option150 is Vlan ID.
subopt151	Enables and sets the DHCP option 82 sub-option151.
subopt151 disable	Disables the DHCP option 82 sub-option151.
subopt151 area-name <WORD/NAME>	Sets the DHCP option 82 sub-option151's Area Name.
subopt151 ssid	Sets the DHCP option 82 sub-option151 is Essid.

## **no option82**

To disable DHCP option 82, use the following command:

```
no option82
```

## **sta-info-extraction**

To enable station information extraction (client fingerprinting), use the following command:

```
sta-info-extraction
```

### **Defaults**

Enabled

## **no sta-info-extraction**

To disable station information extraction (client fingerprinting), use the following command:

```
no sta-info-extraction
```

## **max-clients**

To set the maximum number of clients for a specific WLAN, use the following command:

```
max-clients <NUMBER>
```

### **Syntax Description**

max-clients	Configure the maximum number of clients that the WLAN can support
<NUMBER>	Set the maximum clients to this value

### **Defaults**

100

### **Example**

```
ruckus(config-wlan)# max-clients 100
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **802dot11d**

To enable 802.11d for the WLAN, use the following command:

```
802dot11d
```

### ***Defaults***

Enabled

## **no 802dot11d**

To disable 802.11d for the WLAN, use the following command:

```
no 802dot11d
```

## **application-visibility**

Use the following command to enable application visibility:

```
application-visibility
```

### ***Defaults***

Disabled

### ***Example***

```
ruckus(config-wlan)# application-visibility
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no application-visibility**

Use the following command to disable application visibility:

```
no application-visibility
```

## **apply-policy-group**

Use the following command to apply an application denial policy to the WLAN:

```
apply-policy-group <WORD>
```

### ***Defaults***

None

### **Example**

```
ruckus(config-wlan)# apply-policy-group facebook
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **auto-proxy**

To enable auto-proxy and set the location of the wpad.dat file, use the following command:

```
auto-proxy [<wpad-saved-on-zd | wpad-saved-on-external-  
server>] url <WORD>
```

### **Syntax Description**

auto-proxy	Enable auto-proxy and specify the location of the wpad.dat file
wpad-saved-on-zd	WPAD.DAT file is saved on ZoneDirector
wpad-saved-on-external-server	WPAD.DAT file is saved on an external server
url	Specify the WPAD URL configured on DHCP/DNS server
<WORD>	Auto-proxy path and file name

### **Defaults**

None.

### **Example**

```
ruckus(config-wlan)# auto-proxy wpad-saved-on-zd url 192.168.0.2/  
wpad.dat
```

The file has been loaded into ZoneDirector successfully, Please use 'import' to apply it

```
ruckus(config-wlan)#
```



## **no auto-proxy**

To disable auto-proxy, use the following command:

```
no auto-proxy
```

## **pmk-cache**

To set the PMK cache time to the specified number in minutes (1~720 minutes), use the following command:

```
pmk-cache timeout <NUMBER>
```

## **Defaults**

720 minutes

## **no pmk-cache**

To disable PMK cache, use the following command:

```
no pmk-cache
```

## **pmk-cache-for-reconnect**

To apply PMK cache when client reconnects (default), use the following command:

```
pmk-cache-for-reconnect
```

## **no pmk-cache-for-reconnect**

To disable application of PMK caching when client reconnects, use the following command:

```
no pmk-cache-for-reconnect
```

When “no pmk-cache-for-reconnect” is set, the controller attempts to look up PMK cache for roaming clients only, so every client reconnection requires a full reauthentication. A graceful roaming (disconnect before connecting to the roam-to AP) is not regarded as roaming from the controller’s perspective.

## **Defaults**

Enabled

## **roaming-acct-interim-update**

To enable accounting interim-updates when a client roams, use the following command:

```
roaming-acct-interim-update
```

When “roaming-acct-interim-update” is set, all traffic and session-id data from the original session is carried over to the new session.

### **Defaults**

Disabled.

### **no roaming-acct-interim-update**

To disable accounting interim updates when a client roams (default: disabled), use the following command:

```
no roaming-acct-interim-update
```

### **zero-it-activation**

To enable Zero-IT activation, use the following command:

```
zero-it-activation  
zero-it
```

### **Syntax Description**

---

zero-it-activation	Enable Zero-IT activation
zero-it	Enable Zero-IT activation

---

### **Defaults**

Disabled.

### **Example**

```
ruckus(config-wlan)# zero-it-activation
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **no zero-it-activation**

To disable Zero-IT activation, use the following command:

```
no zero-it-activation  
no zero-it
```

## ***Syntax Description***

---

no zero-it-activation	Disable Zero-IT activation
no zero-it	Disable Zero-IT activation

---

## ***Defaults***

Disabled.

## ***Example***

```
ruckus(config-wlan)# no zero-it
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## **Configuring Dynamic PSKs**

Use the following commands to enable and configure Ruckus Dynamic Pre-Shared Key functionality for the WLAN.

### **dynamic-psk enable**

To enable Dynamic Pre-Shared Keys, use the following command:

```
dynamic-psk enable
```

## ***Syntax Description***

---

dynamic-psk enable	Enable Dynamic PSK
--------------------	--------------------

---

## ***Defaults***

None.

## ***Example***

```
ruckus(config-wlan)# dynamic-psk enable
```

The DPSK can't be enabled or disabled when the wlan type is not Standard Usage and Encryption method is not WPA or WPA2 and Authentication method is not open and Zero-IT is not enabled.

```
ruckus(config-wlan)# zero-it
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)# dynamic-psk enable
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **dynamic-psk passphrase-len**

To set the Dynamic Pre-Shared Key passphrase length, use the following command:

```
dynamic-psk passphrase-len <NUMBER>
```

## **dynamic-psk type**

To sets the type of dynamic PSK (secure or mobile-friendly), use the following command:

```
dynamic-psk type [mobile-friendly|secure]
```

### ***Syntax Description***

dynamic-psk type	Set the DPSK type
mobile-friendly	Set the DPSK type to mobile-friendly
secure	Set the DPSK type to secure

### ***Defaults***

Secure

### ***Example***

```
ruckus(config-wlan)# dynamic-psk type mobile-friendly
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **no dynamic-psk**

To disable Dynamic Pre-Shared Keys on the WLAN, use the following command:

```
no dynamic-psk
```

## **limit-dpsk**

To enable Dynamic PSK limits and set the max number of devices per user, use the following command:

```
limit-dpsk <NUMBER>
```

## **no limit-dpsk**

To disable Dynamic PSK limits, use the following command:

```
no limit-dpsk
```

## **dynamic-psk-expiration**

To set the WLAN Dynamic PSK expiration, use the following command:

```
dynamic-psk-expiration [length|start-point] <WORD>
```

## ***Syntax Description***

dynamic-psk-expiration	Sets the DPSK expiration.
length	Sets the DPSK expiration length.
unlimited	Sets wlan dynamic psk expiration to unlimited.
one-day	Sets wlan dynamic psk expiration to one day.
one-week	Sets wlan dynamic psk expiration to one week.
two-weeks	Sets wlan dynamic psk expiration to two weeks.
one-month	Sets wlan dynamic psk expiration to one month.
two-months	Sets wlan dynamic psk expiration to two months.
three-months	Sets wlan dynamic psk expiration to three months.
half-a-year	Sets wlan dynamic psk expiration to half a year.
one-year	Sets wlan dynamic psk expiration to one year.
two-years	Sets wlan dynamic psk expiration to two years.
start-point	Sets the DPSK validity start-point.
first-use	The D-PSK expiration will be calculated from when it is first used.
creation-time	The D-PSK expiration will be calculated from when it is created.

## ***Example***

```
ruckus(config-wlan)# dynamic-psk-expiration start-point first-use  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-wlan)# dynamic-psk-expiration length one-week  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-wlan)#
```

## **no l2acl**

To disable Layer 2 Access Control Lists, use the following command:

```
no l2acl
```

## **no role-based-access-ctrl**

To disable role based access control policy service, use the following command:

```
no role-based-access-ctrl
```

## **no l3acl**

To disable Layer 3/4 ACLs, use the following command:

```
no l3acl
```

## **no l3acl-ipv6**

To disable Layer 3/4 IPv6 ACLs, use the following command:

```
no l3acl-ipv6
```

## **no vlanpool**

To disable the VLAN pool for this WLAN, use the following command:

```
no vlanpool
```

## **no dvcpcy**

To disable device policy for this WLAN, use the following command:

```
no dvcpcy
```

## **rate-limit**

To set the rate limiting for the WLAN, use the following command:

```
rate-limit uplink <NUMBER> downlink <NUMBER>
```

### ***Syntax Description***

rate-limit	Set the rate limit
uplink	Set the uplink rate limit
downlink	Set the downlink rate limit
<NUMBER>	Set the rate limiting to the value specified.

### ***Defaults***

None.

### **Example**

```
ruckus(config-wlan)# rate-limit uplink 20 downlink 20
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

### **no rate-limit**

To disable the rate limit, use the following command:

```
no rate-limit
```

### **Syntax Description**

---

no rate-limit	Disable rate limiting for the WLAN
---------------	------------------------------------

---

### **Defaults**

Disabled.

### **Example**

```
ruckus(config-wlan)# no rate-limit
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **vlanpool**

To configure a VLAN pool with the specified name, use the following command:

```
vlanpool <WORD>
```

### **no mac-addr-format**

Sets MAC auth username and password to format aabbccddeeff.

### **mac-addr-format**

Sets MAC auth username and password to one of the following formats:

---

mac-addr-format aa-bb-cc-dd-ee- ff	Sets MAC auth username and password to format aa-bb-cc-dd-ee-ff.
--	--

---



mac-addr-format aa:bb:cc:dd:ee:ff	Sets MAC auth username and password to format aa:bb:cc:dd:ee:ff.
mac-addr-format AABBCCDDEEFF	Sets MAC auth username and password to format AABBCCDDEEFF.
mac-addr-format AA-BB-CC-DD-EE-FF	Sets MAC auth username and password to format AA-BB-CC-DD-EE-FF.
mac-addr-format AA:BB:CC:DD:EE:F F	Sets MAC auth username and password to format AA:BB:CC:DD:EE:FF.

### **acl dvccpy**

To apply a Device Policy to the WLAN, use the following command:

```
acl dvccpy <WORD>
```

### **acl prece**

To apply a Precedence Policy to the WLAN, use the following command:

```
acl prece <WORD>
```

### **acl role-based-access-ctrl**

To enable Role based Access Control Policy on the WLAN, use the following command:

```
acl role-based-access-ctrl
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-wlan)# acl role-based-access-ctrl
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlan)#
```

## **qos classification**

To enable Quality of Service classification, use the following command:

```
qos classification
```

## **no qos classification**

To disable Quality of Service classification, use the following command:

```
no qos classification
```

## **qos heuristics-udp**

To enable QoS heuristics for UDP traffic, use the following command:

```
qos heuristics-udp
```

## **no qos heuristics-udp**

To disable QoS heuristics for UDP traffic, use the following command:

```
no qos heuristics-udp
```

## **qos directed-multicast**

To enable QoS directed multicast, use the following command:

```
qos directed-multicast
```

## **no qos directed-multicast**

To disable QoS directed multicast, use the following command:

```
no qos directed-multicast
```

## **qos igmp-snooping**

To disable QoS directed multicast, use the following command:

```
qos igmp-snooping
```

## **no qos igmp-snooping**

To disable QoS IGMP snooping, use the following command:

```
no qos igmp-snooping
```

## **qos mld-snooping**

To enable QoS MLD snooping, use the following command:

```
no qos mld-snooping
```

## **no qos mld-snooping**

To disable QoS MLD snooping, use the following command:

```
no qos mld-snooping
```

## **qos tos-classification**

To enable QoS TOS classification, use the following command:

```
qos tos-classification
```

## **no qos tos-classification**

To disable QoS TOS classification, use the following command:

```
no qos tos-classification
```

## **qos priority high**

To set QoS priority to 'high', use the following command:

```
qos priority high
```

## **qos priority low**

To set QoS priority to 'low', use the following command:

```
qos priority low
```

## **qos directed-threshold**

To set the QoS directed threshold, use the following command:

```
qos directed-threshold <NUMBER>
```

## **disable-dgaf**

To disable Downstream Group-Address Frame Forwarding, use the following command (Hotspot 2.0 WLAN only):

```
disable-dgaf
```

## **no disable-dgaf**

To enable Downstream Group-Address Frame Forwarding, use the following command (Hotspot 2.0 WLAN only):

```
no disable-dgaf
```

## **proxy-arp**

To enable Proxy ARP service for the WLAN, use the following command:

```
proxy-arp
```

### **no proxy-arp**

To disable Proxy ARP service for the WLAN, use the following command:

```
no proxy-arp
```

### **80211w-pmf**

To enable 802.11w PM, use the following command:

```
80211w-pmf
```

### **no 80211w-pmf**

To disable 802.11w PMF, use the following command:

```
no 80211w-pmf
```

### **ignor-unauth-stats**

To enable ignoring unauthorized client statistics, use the following command:

```
ignor-unauth-stats
```

### **no ignor-unauth-stats**

To disable ignoring unauthorized client statistics, use the following command:

```
no ignor-unauth-stats
```

### **show**

To display the WLAN settings, use the following command:

```
show
```

### ***Syntax Description***

---

show	Display WLAN settings
------	-----------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config)# wlan ruckus1
```

The WLAN service 'ruckus1' has been loaded. To save the WLAN service, type 'end' or 'exit'.

```
ruckus(config-wlan)# show
```

WLAN Service:

ID:

1:

```
NAME = Ruckus-Wireless-1
Tx. Rate of Management Frame(2.4GHz) = 2.0Mbps
Tx. Rate of Management Frame(5GHz) = 6.0Mbps
Beacon Interval = 100ms
SSID = Ruckus-Wireless-1
Description = Ruckus-Wireless-1
Type = Standard Usage
Authentication = open
Encryption = wpa
Algorithm = aes
Passphrase = password
FT Roaming = Disabled
802.11k Neighbor report = Disabled
Web Authentication = Disabled
Authentication Server = Disabled
Accounting Server = Disabled
Called-Station-Id type = wlan-bssid
Tunnel Mode = Disabled
DHCP relay = Disabled
Max. Clients = 100
Isolation per AP = Disabled
Isolation across AP = Disabled
Zero-IT Activation = Enabled
Load Balancing = Disabled
Band Balancing = Disabled
Dynamic PSK = Enabled
Dynamic PSK Passphrase Length =
Limit Dynamic PSK = Disabled
Auto-Proxy configuration:
    Status = Disabled
Inactivity Timeout:
    Status = Disabled
VLAN-ID = 1
```

```
Dynamic VLAN = Disabled
Closed System = Disabled
OFDM-Only State = Disabled
Multicast Filter State = Disabled
802.11d State = Disabled
Force DHCP State = Disabled
Force DHCP Timeout = 0
DHCP Option82:
    Status = Disabled
    Option82 sub-Option1 = Disabled
    Option82 sub-Option2 = Disabled
    Option82 sub-Option150 = Disabled
    Option82 sub-Option151 = Disabled
Ignore unauthorized client statistic = Disabled
STA Info Extraction State = Enabled
BSS Minrate = Disabled
Call Admission Control State = Disabled
PMK Cache Timeout= 720 minutes
PMK Cache for Reconnect= Enabled
NAS-ID Type= wlan-bssid
Roaming Acct-Interim-Update= Disabled
PAP Message Authenticator = Enabled
Send EAP-Failure = Disabled
L2/MAC = No ACLS
L3/L4/IP Address = No ACLS
L3/L4/IPv6 Address = No ACLS
Precedence = No ACLS
Proxy ARP = Disabled
Device Policy = No ACLS
Role based Access Control Policy = Disabled
SmartRoam = Disabled  Roam-factor = 1
White List = No ACLS
Application Visibility = disabled
Apply Policy Group = No_Denys
```

```
ruckus(config)#
```

# Configure WLAN Group Settings Commands

Use the `wlan-group` commands to configure the settings of a particular WLAN group.

## wlan-group

To create a new WLAN group or update an existing WLAN group, use the following command:

```
wlan-group <WORD>
```

### Syntax Description

---

<code>wlan-group</code>	Configure the WLAN group
<code>&lt;WORD&gt;</code>	Name of the WLAN group

---

### Defaults

Default.

### Example

```
ruckus# config
ruckus(config)# wlan-group wlangroup2
The WLAN group 'wlangroup2' has been created. To save the WLAN
group, type 'end' or 'exit'.
ruckus(config-wlangrp)#
```

## no wlan-group

To delete a WLAN group from the list, use the following command:

```
no wlan-group <WORD>
```

### Syntax Description

---

<code>no wlan-group</code>	Delete the WLAN group
<code>&lt;WORD&gt;</code>	Name of the WLAN group

---

### Defaults

None.

### **Example**

```
ruckus(config)# no wlan-group wlan-grp-01
The WLAN group 'wlan-grp-01' has been removed.
ruckus(config)#
```

### **abort**

To exit the `wlan-group` context without saving changes, use the `abort` command. Enter this command from within the context of the WLAN group that you are configuring.

```
abort
```

### **Syntax Description**

---

<code>abort</code>	Exit the WLAN group without saving changes
--------------------	--

---

### **Defaults**

None.

### **Example**

```
ruckus# config
ruckus(config)# wlan-group wlangroup2
The WLAN group 'wlangroup2' has been created. To save the WLAN
group, type 'end' or 'exit'.
ruckus(config-wlangrp)# abort
No changes have been saved.
ruckus(config)#
```

### **end**

To save changes to the WLAN group settings and exit the `wlan-group` context, use the following command. Enter this command from within the context of the WLAN group that you are configuring.

```
end
```



## **Syntax Description**

---

<code>end</code>	Save changes, and then exit the WLAN group
------------------	--

---

## **Defaults**

None.

## **Example**

```
ruckus# config
ruckus(config)# wlan-group wlangroup2
The WLAN group 'wlangroup2' has been created. To save the WLAN
group, type 'end' or 'exit'.
ruckus(config-wlangrp)# end
The WLAN group 'wlangroup2' has been updated.
Your changes have been saved.
ruckus(config)#
```

## **exit**

To save changes to the WLAN group settings and exit the `wlan-group` context, use the `exit` command. Enter this command from within the context of the WLAN group that you are configuring.

```
exit
```

## **Syntax Description**

---

<code>exit</code>	Save changes, and then exit the WLAN group
-------------------	--

---

## **Defaults**

None.

## **Example**

```
ruckus# config
ruckus(config)# wlan-group wlangroup2
```

The WLAN group entry 'wlangroup2' has been loaded. To save the WLAN group, type 'end' or 'exit'.

```
ruckus(config-wlangrp)# exit
```

The WLAN group 'wlangroup2' has been updated.

Your changes have been saved.

```
ruckus(config)#
```

## **quit**

To exit the `wlan-group` context without saving changes, use the following command. Enter this command from within the context of the WLAN group that you are configuring.

```
quit
```

## **Syntax Description**

---

<code>quit</code>	Exit the WLAN group without saving changes
-------------------	--

---

## **Defaults**

None.

## **Example**

```
ruckus# config
```

```
ruckus(config)# wlan-group wlangroup2
```

The WLAN group entry 'wlangroup2' has been loaded. To save the WLAN group, type 'end' or 'exit'.

```
ruckus(config-wlangrp)# quit
```

No changes have been saved.

```
ruckus(config)#
```

## **name**

To set the WLAN group name, use the following command. Enter this command from within the context of the WLAN group that you are configuring.

```
name <WORD>
```

## Syntax Description

---

name	Configure the WLAN group name
<WORD>	Set the WLAN group name to this value

---

## Defaults

None.

## Example

```
ruckus# config
ruckus(config)# wlan-group wlangroup2
The WLAN group entry 'wlangroup2' has been loaded. To save the WLAN
group, type 'end' or 'exit'.
ruckus(config-wlangrp)# name wlangroup2
ruckus(config-wlangrp)# show
WLAN Group:
  ID:
  2:
    Name= wlangroup2
    Description=
    WLAN Service:

ruckus(config-wlangrp)#
```

## description

To set the WLAN group description, use the following command. Enter this command from within the context of the WLAN group that you are configuring. Multiple word text must be enclosed in quotes.

```
description <WORD>
```

## Syntax Description

---

description	Configure the WLAN group description
<WORD>	Set the WLAN group description to this value

---

## Defaults

None.

## Example

```
ruckus# config
ruckus(config)# wlan-group wlangroup2
ruckus(config-wlangrp)# description "WLAN Group 2"
ruckus(config-wlangrp)# show
WLAN Group:
  ID:
    2:
      Name= wlangroup2
      Description= WLAN Group 2
      WLAN Service:

ruckus(config-wlangrp)#
```

## wlan

To add a WLAN service to the WLAN group, use the following command. Enter this command from within the context of the WLAN group that you are configuring.

```
wlan <WORD>
```

## Syntax Description

---

wlan	Add a WLAN to the WLAN group
<WORD>	Name of the WLAN to be added

---

## Defaults

None.

## Example

```
rruckus(config-wlangrp)# wlan ruckus1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlangrp) # show
WLAN Group:
  ID:
  :
  Name= wlangroup1
  Description=
  WLAN Service:
    WLAN1:
      NAME= ruckus1
      VLAN=

ruckus(config-wlangrp) #
```

## **no wlan**

To remove a WLAN service from the WLAN group, use the following command. Enter this command from within the context of the WLAN group that you are configuring.

```
no wlan <WORD>
```

## ***Syntax Description***

<code>no wlan</code>	Delete an existing WLAN service from the WLAN group
<code>&lt;WORD&gt;</code>	Name of the WLAN to be removed

## ***Defaults***

None.

## ***Example***

```
ruckus(config-wlangrp) # no wlan ruckus1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-wlangrp) #
```

## wlan vlan override none

To add a WLAN service to the WLAN group and set the VLAN tag to 'No Change', use the following command. Enter this command from within the context of the WLAN group that you are configuring.

```
wlan <WORD> vlan override none
```

### Syntax Description

---

wlan <WORD>	Add the WLAN to the WLAN group
wlan override none	Set the VLAN tag to No Change

---

### Defaults

None.

### Example

```
ruckus(config-wlangrp)# wlan ruckus1 vlan override none  
The command was executed successfully. To save the changes, type  
'end' or 'exit'.  
ruckus(config-wlangrp)#
```

## wlan vlan override tag

To add a WLAN service to the WLAN group and set the VLAN tag to the specified VLAN ID, use the following command:

```
wlan <NAME> vlan override tag <NUMBER>
```

### Syntax Description

---

wlan <NAME>	Add the <NAME> to the WLAN group
wlan override tag <NUMBER>	Set the VLAN tag of <NAME> to the specified <NUMBER>

---

### Defaults

None.

### Example

```
ruckus(config-wlangrp)# wlan ruckus1 vlan override tag 12
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-wlangrp)#
```

### show

To display WLAN group settings, use the following command:

```
show
```

### Defaults

```
ruckus(config-wlangrp)# show
```

WLAN Group:

ID:

1:

Name= Default

Description= Default WLANs for Access Points

WLAN Service:

WLAN1:

NAME= Ruckus1

VLAN=

```
ruckus(config-wlangrp)#
```

## Configure Role Commands

Use the `role` commands to configure user roles on the controller. To run these commands, you must first enter the `config-role` context.

### role

To create a new role or modify an existing role, use the following command:

```
role <WORD>
```

## Syntax Description

role	Create or modify a user role
<WORD>	Name of role

## Defaults

None.

## Example

```
ruckus(config)# role role1
The role entry 'role1' has been created
ruckus(config-role)#
```

## no role

To delete a role entry from the list, use the following command:

```
no role <WORD>
```

## Syntax Description

no role	Delete a user role
<WORD>	Name of role

## Defaults

None.

## Example

```
ruckus(config)# no role role1
The Role 'role1' has been deleted.
ruckus(config)#
```

## abort

To exit the `config-role` context without saving changes, use the `abort` command. Enter this command from within the context of the role that you are configuring.



```
abort
```

### **Syntax Description**

---

abort	Exit the role without saving changes
-------	--------------------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# abort
No changes have been saved.
ruckus(config)#
```

### **end**

To save changes, and then exit the `config-role` context, use the following command:

```
end
```

### **Syntax Description**

---

end	Save changes, and then exit the context
-----	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# end
The Role entry has saved successfully.
Your changes have been saved.
ruckus(config)#
```

### **exit**

To save changes, and then exit the `config-role` context, use the following command:

```
exit
```

## **Syntax Description**

---

exit	Save changes, and then exit the context
------	---

---

## **Defaults**

None.

## **Example**

```
ruckus(config-role)# exit
The Role entry has saved successfully.
Your changes have been saved.
ruckus(config)#
```

## **quit**

To exit the `config-role` context without saving changes, use the `quit` command. Enter this command from within the context of the role that you are configuring.

```
quit
```

## **Syntax Description**

---

quit	Exit the role without saving changes
------	--------------------------------------

---

## **Defaults**

None.

## **Example**

```
ruckus(config-role)# quit
No changes have been saved.
ruckus(config)#
```

## **name**

To set the name of a user role, use the following command:

```
name <WORD>
```

### **Syntax Description**

name	Set the name of a user role
<WORD>	Set to this role

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# name guest33
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **description**

To set the description for a user role, use the following command:

```
description <WORD>
```

### **Syntax Description**

description	Set the description of a user role
<WORD>	Set to this description

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# description testforCLI
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **group-attributes**

To set the group attributes of a user role, use the following command:

```
group-attributes <WORD>
```

### **Syntax Description**

group-attributes	Set the attributes of a user role
<WORD>	Set to this attribute

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# group-attributes ruckus1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **wlan-allowed**

To set the WLANs to which a user role will have access, use the following command:

```
wlan-allowed [all | specify-wlan]
```

### **Syntax Description**

wlan-allowed	Set the WLANs to which a role will have access
all	Grant access to all WLANs
specify-wlan	Grant access to a specific WLAN

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# wlan-allowed all
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-role)# wlan-allowed specify-wlan
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## no specify-wlan-access

To remove a particular WLAN from the list of WLANs that a user role can access, use the following command:

```
no specify-wlan-access <WORD/SSID>
```

### Syntax Description

no specify-wlan-access	Remove access to a WLAN by a user role
<WORD/SSID>	Remove access to this WLAN

### Defaults

None.

### Example

```
ruckus(config-role)# no specify-wlan-access joejoe98
The wlan 'joejoe98' has been removed from the Role.
```

## specify-wlan-access

To add a particular WLAN to the list of WLANs that a user role can access, use the following command:

```
specify-wlan-access <wlan_ssid>
```

### Syntax Description

specify-wlan-access	Add access to a WLAN by a user role
<wlan_ssid>	Add access to this WLAN

### Defaults

None.

### Example

```
ruckus(config-role)# specify-wlan-access joejoe98
The wlan 'joejoe98' has been added to the Role.
```

## no guest-pass-generation

To remove guest pass generation privileges from a user role, use the following command:

```
no guest-pass-generation
```

### **Syntax Description**

---

no guest-pass-generation	Remove guest pass generation privileges from a user role
--------------------------	--

---

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# no guest-pass-generation
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## guest-pass-generation

To add guest pass generation privileges to a user role, use the following command:

```
guest-pass-generation
```

### **Syntax Description**

---

guest-pass-generation	Add guest pass generation privileges to a user role
-----------------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# guest-pass-generation
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## no admin

To remove ZoneDirector administration privileges from a user role, use the following command:

```
no admin
```

### **Syntax Description**

---

no admin	Remove ZoneDirector administration privileges from a user role
----------	--

---

### **Defaults**

None.

### **Example**

```
ruckus(config-role)# no admin
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## admin

To add ZoneDirector administration privileges to a user role, use the following command:

```
admin [super | operator | monitoring]
```

### **Syntax Description**

---

admin	Add ZoneDirector administration privileges to a user role
super	Sets to Super (Perform all configuration and management tasks)
operator	Sets to Operator (Change settings affecting single AP's only)
monitoring	Sets to Monitoring (Monitoring and viewing operation status only)

---

## Defaults

None.

## Example

```
ruckus(config-role)# admin super
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## access-ctrl

Enables access control policy.

## Defaults

Disabled

## Example

```
ruckus(config)# role role1
```

The Role entry 'role1' has been created.

```
ruckus(config-role)# access-ctrl
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-role)# show
```

Role:

ID:

:

Name= role1

Description=

Group Attributes=

Guest Pass Generation= Disallowed

ZoneDirector Administration:

Status= Disallowed

Allow All WLANs:

Mode= Allow Specify WLAN access

Access Control Policy= Allowed

Allow All OS Types:

Mode= Allow all OS types to access

VLAN = Any



```
Rate Limiting Uplink = Disabled  
Rate Limiting Downlink = Disabled
```

```
ruckus(config-role)#
```

### **no access-ctrl**

Disables access control policy.

```
no access-ctrl
```

### **os-type-allowed all**

Allows all OS types to access.

```
os-type-allowed all
```

### **os-type-allowed specify**

Specifies OS types access.

```
os-type-allowed specify
```

### **specify-os-type-access**

Adds the specify OS type into the role entry.

```
specify-os-type-access <WORD>
```

### **Defaults**

None

### **Example**

```
ruckus(config)# role role1
```

The Role entry 'role1' has been created.

```
ruckus(config-role)# access-ctrl
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-role)# os-type-allowed specify
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-role)# specify-os-type-access Windows
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-role)#
```

### **no specify-os-type-access**

Deletes the specify OS type from the role entry.

```
no specify-os-type-access <WORD>
```

### **vlan**

Sets the VLAN ID to the specified ID number or "none"

```
vlan <NUMBER>
```

### **rate-limit uplink**

Sets the rate limiting of uplink.

```
rate-limit uplink <NUMBER>
```

### **rate-limit uplink downlink**

Sets the rate limiting of downlink.

```
rate-limit uplink <NUMBER> downlink <NUMBER>
```

### **no rate-limit**

Sets rate limiting to Disable.

```
no rate-limit
```

### **show**

To display the settings of a role, use the following command:

```
show
```

### ***Syntax Description***

---

show	Display the settings of a role
------	--------------------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-role)# show
```

```

Role:
  ID:
  :
  Name= role1
  Description=
  Group Attributes=
  Guest Pass Generation= Disallowed
  ZoneDirector Administration:
    Status= Disallowed
  Allow All WLANs:
    Mode= Allow Specify WLAN access

ruckus(config-role)#

```

## Configure VLAN Pool Commands

Use the vlan-pool commands to create and configure a VLAN pool. Running these commands enters the config-vlan-pool context from within the config context.

### vlan-pool

To create a new VLAN pool or modify an existing pool, and enter the config-vlan-pool context, use the following command:

```
vlan-pool <WORD>
```

### Syntax Description

abort	Exits the config-vlanpool context without saving changes.
end	Saves changes, and then exits the config-vlanpool context.
exit	Saves changes, and then exits the config-vlanpool context.
quit	Exits the config-vlanpool context without saving changes.
name <WORD>	Sets the vlan pool entry name.

description <WORD>	Sets the vlan pool entry description.
vlan	Adds or deletes vlans from the vlan pool.
vlan add <WORD>	Add the vlan to the specified pool.
vlan delete <WORD>	Delete the vlan from the specified pool.
vlan show	
option <NUMBER>	Set the option 1 'Mac Hash' 2 'Round-Robin' 3 'Least-Used' to the specified pool.
show	Displays pool settings.

### **Example**

```
ruckus(config)# vlan-pool vlan-pool-1
The vlan pool entry 'vlan-pool-1' has been created.
ruckus(config-vlanpool)# description "vlan pool for printers"
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-vlanpool)# option 1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-vlanpool)# vlan add 10
ruckus(config-vlanpool)# vlan add 20
ruckus(config-vlanpool)# vlan add 30
ruckus(config-vlanpool)# vlan add 50-56
ruckus(config-vlanpool)# show
VLAN Pool:
  ID:
  :
  Name = vlan-pool-1
  Description = vlan pool for printers
  Option = 1
  VLANSSET = 10,20,30,50-56

ruckus(config-vlanpool)# end
The vlan pool entry has saved successfully.
Your changes have been saved.
ruckus(config)#
```

## no vlan-pool

To delete a VLAN pool, use the following command:

```
no vlan-pool <WORD>
```

### Example

```
ruckus(config)# no vlan-pool vlan-pool-1
The vlan pool 'vlan-pool-1' has been deleted.
ruckus(config)#
```

## Configure User Commands

Use the `user` commands to configure a user's name, password, and role. To run these commands, you must first enter the `config-user` context.

### user

To create a user or modify an existing user and enter the `config-user` context, use the following command:

```
user <WORD>
```

### Syntax Description

<code>user</code>	Create or modify a user entry
<code>&lt;WORD&gt;</code>	Name of the user

### Defaults

None.

### Example

```
ruckus(config)# user johndoe
The User entry 'johndoe' has been created.
ruckus(config-user)#
```

### no user

To delete a user record, use the following command:

```
no user <WORD>
```

### **Syntax Description**

user	Create or modify a user entry
<WORD>	Name of the user

### **Defaults**

None.

### **Example**

```
ruckus(config)# no user johndoe
The User 'johndoe' has been deleted.
ruckus(config)#
```

### **abort**

To exit the `config-user` context without saving changes, use the `abort` command. Enter this command from within the context of the user that you are configuring.

```
abort
```

### **Syntax Description**

abort	Exit the user settings without saving changes
-------	---

### **Defaults**

None.

### **Example**

```
ruckus(config-user)# abort
No changes have been saved.
ruckus(config)#
```

## end

To save changes, and then exit the `config-user` context, use the following command (you must first set a password before exiting):

```
end
```

### **Syntax Description**

---

end	Save changes, and then exit the context
-----	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-user)# end
The User entry has saved successfully.
Your changes have been saved.
ruckus(config)#
```

## exit

To save changes, and then exit the `config-user` context, use the following command (you must first set a password before exiting):

```
exit
```

### **Syntax Description**

---

exit	Save changes, and then exit the context
------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-user)# exit
The User entry has saved successfully.
Your changes have been saved.
```

```
ruckus(config)#
```

## quit

To exit the `config-user` context without saving changes, use the `quit` command. Enter this command from within the context of the user that you are configuring.

```
quit
```

## Syntax Description

---

quit	Exit the user settings without saving changes
------	---

---

## Defaults

None.

## Example

```
ruckus(config-role)# quit  
No changes have been saved.  
ruckus(config)#
```

## user-name

To set the name of a user, use the following command:

```
user-name <WORD>
```

## Syntax Description

---

user-name	Set the name of a user
<WORD>	Set to this user name

---

## Defaults

None.

## Example

```
ruckus(config-user)# user-name joel
```



The command was executed successfully. To save the changes, type 'end' or 'exit'.

## full-name

To set the full name of a user, use the following command:

```
full-name <WORD>
```

### Syntax Description

full-name	Set the full name of a user
<WORD>	Set to this full name

### Defaults

None.

### Example

```
ruckus(config-user)# full-name joejoe
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## password

To set the password of a user, use the following command:

```
password <WORD>
```

### Syntax Description

password	Set the password of a user
<WORD>	Set to this password

### Defaults

None.

### Example

```
ruckus(config-user)# password 1234
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## role

To assign a role to a user, use the following command:

```
role <WORD>
```

### Syntax Description

---

role	Assign a role to a user
<WORD>	Assign this role

---

### Defaults

Default

### Example

```
ruckus(config-user)# role guest
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## show

To display the settings of a user, use the following command:

```
show
```

### Syntax Description

---

show	Show user settings
------	--------------------

---

### Defaults

None.

### Example

```
ruckus(config-user)# show
```

```
User:
```

```
ID:
```

```
:  
User Name= joe1  
Full Name= joejoe  
Password= 1234  
Role= guest
```

## Configure Guest Access Commands

Use the `guest-access` commands to configure guest access services. To run these commands, you must first enter the `config-guest-access` context.

### guest-access

To create/configure a Guest Access service and enter the `config-guest-access` context, use the following command:

```
guest-access <WORD>
```

#### *Example*

```
ruckus(config)# guest-access guestpolicy1  
The Guest Access entry 'guestpolicy1' has been created.  
ruckus(config-guest-access)#
```

### no guest-access

To delete a Guest Access service, use the following command:

```
no guest-access
```

#### *Example*

```
ruckus(config)# no guest-access guest1  
The Guest Access 'guest1' has been deleted.  
ruckus(config)#
```

### abort

To exit the `config-guest-access` context without saving changes, use the `abort` command.

```
abort
```

**end**

To save changes, and then exit the `config-guest-access` context, use the following command:

```
end
```

**exit**

To save changes, and then exit the `config-guest-access` context, use the following command:

```
exit
```

**quit**

To exit the `config-guest-access` context without saving changes, use the `quit` command.

```
quit
```

**name**

To set the name of the guest access policy, use the following command:

```
name <WORD>
```

**self-service**

To enable guest pass self-registration, use the following command:

```
self-service
```

**no self-service**

To disable guest pass self-registration, use the following command:

```
no self-service
```

**guestpass-duration**

To set the guest pass duration, use the following command:

```
guestpass-duration [hour|day|week] <NUMBER>
```

**guestpass-reauth**

To set the guest pass reauthorization timeout, use the following command:

```
guestpass-reauth [min|hour|day|week] <NUMBER>
```

## **no guestpass-reauth**

To disable guest pass reauthorization timeout, use the following command:

```
no guestpass-reauth
```

## **guestpass-share-number**

To set the limit on how many devices can share one guest pass, use the following command (valid values: [0, 10] and 0 means unlimited):

```
guestpass-share-number <NUMBER>
```

## **guestpass-sponsor**

To enable guest pass sponsor approval, use the following command:

```
guestpass-sponsor
```

## **no guestpass-sponsor**

To disable guest pass sponsor approval, use the following command:

```
no guestpass-sponsor
```

## **guestpass-sponsor-auth-server**

Sets the authentication server to 'Local Database' or to a specified AAA server name, use the following command:

```
guestpass-sponsor-auth-server [local|name <WORD>]
```

## **guestpass-sponsor-number**

To set the number of sponsors that can be used for this guest pass service (valid values: [1,5]), use the following command:

```
guestpass-sponsor-number <NUMBER>
```

## **guestpass-notification**

To set the notification method for delivering guest passes, use the following command:

```
guestpass-notification <NUMBER>
```

### ***Options***

1-Device Screen

2-Mobile

3-Email

## 4-Mobile and Email

### **guestpass-terms-and-conditions**

To enable and set the terms and conditions, use the following command:

```
guestpass-terms-and-conditions <WORD>
```

### **no guestpass-terms-and-conditions**

To disable the terms and conditions, use the following command:

```
no guestpass-terms-and-conditions
```

### **onboarding**

To configure onboarding portal options, use the following command:

```
onboarding [key-and-zeroit|zeroit]
```

### ***Syntax Description***

onboarding	Enable onboarding portal.
key-and-zeroit	Enables guest pass and zero-it activation.
zeroit	Enables zero-it activation only.

### ***Defaults***

Enabled, Guest Pass and Zero-IT.

### ***Example***

```
ruckus(config-guest-access)# onboarding key-and-zeroit  
The command was executed successfully.  
ruckus(config-guest-access)#
```

### **no onboarding**

To disable the onboarding portal, use the following command:

```
no onboarding
```

### **no authentication**

To disable guest access authentication, use the following command:

```
no authentication
```

### ***Syntax Description***

---

no authentication	Disable guest access authentication
-------------------	-------------------------------------

---

### ***Defaults***

Enabled.

### ***Example***

```
ruckus(config-guest-access)# no authentication  
The command was executed successfully.
```

### **authentication guest-pass**

To enable guest pass authentication for this guest access service, use the following command:

```
authentication guest-pass
```

### ***Syntax Description***

---

authentication guest-pass	Enable guest pass authentication
---------------------------	----------------------------------

---

### ***Example***

```
ruckus(config-guest-access)# authentication guest-pass  
The command was executed successfully.
```

### **no term-of-use**

To hide the Terms of Use text on the guest pass access page, use the following command:

```
no term-of-use
```

## Syntax Description

---

no term-of-use	Hide Terms of Use
----------------	-------------------

---

### Defaults

Disabled.

### Example

```
ruckus(config-guest-access)# no term-of-use  
The command was executed successfully.
```

### term-of-use

To display and specify the Terms of Use text on the guest pass access page, use the following command:

```
term-of-use <WORD>
```

## Syntax Description

---

term-of-use	Display Terms of Use
<WORD>	Display this text as content of Terms of Use on Guest Pass access page

---

### Defaults

Disabled.

### Example

```
ruckus(config-guest-access)# term-of-use test.guest  
The command was executed successfully.
```

### redirect

To set the URL to which to redirect a guest user after passing authentication, use the following command:

```
redirect [original | url <WORD>]
```



## Syntax Description

---

redirect	Set the URL to which the guest user will be redirected
original	Redirect user to the original page that he intended to visit
url <WORD>	Redirect user to a different URL. Specify the URL in <WORD>.

---

## Defaults

original

## Example

```
ruckus(config-guest-access)# redirect url http://www.ruckuswireless.com
```

The command was executed successfully.

## welcome-text

To configure the text to display on the guest access user login page, use the following command:

```
welcome-text <WORD>
```

## Syntax Description

---

welcome-text	Configure the welcome message
<WORD>	Use this as the welcome message

---

## Defaults

Welcome to the Guest Access login page.

## Example

```
ruckus(config-guest-access)# welcome-text "Welcome to the Guest Access Login Page."
```

The command was executed successfully.

```
ruckus(config-guest-access)#
```

## show

To display the guest access policy settings, use the following command:

```
show
```

### ***Syntax Description***

---

show	Display the guest access settings
------	-----------------------------------

---

### ***Example***

```
ruckus(config-guest-access)# show
Guest Access:
  Name = guestservice1
  Onboarding Portal:
    Aspect = Guest pass and ZeroIT
  Authentication:
    Mode = Use guest pass authentication
    Multiple users to share a single guest pass = Disallowed
  Title = Welcome to the Guest Access login page.
  Terms of Use:
    Status = Disabled
  Redirection:
    Mode = To the URL that the user intends to visit
  Restricted Subnet Access:
    Rules:
      1:
        Description=
        Type= Deny
        Destination Address= local
        Destination Port= Any
        Protocol= Any
      2:
        Description=
        Type= Deny
        Destination Address= 10.0.0.0/8
        Destination Port= Any
        Protocol= Any
      3:
```

```
Description=  
Type= Deny  
Destination Address= 172.16.0.0/12  
Destination Port= Any  
Protocol= Any  
4:  
Description=  
Type= Deny  
Destination Address= 192.168.0.0/16  
Destination Port= Any  
Protocol= Any
```

Restricted IPv6 Access:

Rules:

```
1:  
Description=  
Type= Deny  
Destination Address= local  
Destination Port= Any  
Protocol= Any  
ICMPv6 Type= Any
```

```
ruckus(config-guest-access)#
```

## Configuring Guest Access Restriction Rules

Use the following commands to configure restricted access rules for a guest policy. To use these commands, you must enter the `config-guest-restrict-access` context from within the `config-guest-access` context.

### **no restrict-access-order**

To delete a restrict access order, use the following command:

```
no restrict-access-order <NUMBER>
```

## Syntax Description

no restrict-access-order	Delete a restrict access order
<NUMBER>	Delete this order ID

### Example

```
ruckus(config-guest-access)# no restrict-access-order 4
```

The Restricted Subnet Access entry has been removed from the Guest Access.

```
ruckus(config-guest-access)#
```

## restrict-access-order

To create a new restrict access order or modify an existing restrict access order, use the following command:

```
restrict-access-order <NUMBER>
```

This command enters the config-guest-restrict-access context. The following commands are available from within this context:

## Syntax Description

help	Shows available commands
history	Shows a list of previously run commands.
abort	Exits the config-guest-restrict-access context without saving changes.
end	Saves changes, and then exits the config-guest-restrict-access context.
exit	Saves changes, and then exits the config-guest-restrict-access context.
quit	Exits the config-guest-restrict-access context without saving changes.
order <NUMBER>	Sets the guest access rule order.
description <WORD>	Sets the guest access rule description.

type [allow   deny]	Sets the guest access rule type to allow or deny.
destination [address <ADDR>   port <NUMBER/WORD>]	Sets the destination address/port of a guest access rule.
protocol <NUMBER/WORD>	Sets the protocol of a guest access rule.
show	Displays restricted subnet access settings.

## show

To display guest access restriction settings, use the following command:

```
show
```

### **Syntax Description**

show	Display guest access restriction settings
------	---

### **Defaults**

None.

## order

To configure the guest access rule order, use the following command:

```
order <NUMBER>
```

### **Syntax Description**

order	Set the order of a guest access rule
<NUMBER>	Assign the rule this order

### **Example**

```
ruckus(config-guest-restrict-access)# order 3
```

The command was executed successfully.

## description

To set the description of a guest access rule, use the following command:

```
description <WORD>
```

### **Syntax Description**

description	Set the description of a guest access rule
<WORD>	Set this as description

### **Defaults**

None.

### **Example**

```
ruckus(config-guest-restrict-access)# description guestd3
The command was executed successfully.
```

### **type allow**

To set the guest access rule type to 'allow', use the following command:

```
type allow
```

### **Syntax Description**

type	Set the guest access rule type
allow	Set the rule type to 'allow'

### **Defaults**

Deny.

### **Example**

```
ruckus(config-guest-restrict-access)# type allow
The command was executed successfully.
```

### **type deny**

To set the guest access rule type to 'deny', use the following command:

```
type deny
```

## Syntax Description

type	Set the guest access rule type
deny	Set the rule type to 'deny'

## Defaults

Deny.

## Example

```
ruckus(config-guest-restrict-access)# type deny
```

The command was executed successfully.

## destination address

To set the destination address of the rule, use the following command:

```
destination address <IP-ADDR/WORD>
```

## Syntax Description

destination address	Set the destination address of the rule
IP-ADDR/WORD	Set the destination to this IP address

## Defaults

Any.

## Example

```
ruckus(config-guest-restrict-access)# destination address  
192.168.0.20/24
```

The command was executed successfully.

## destination port

To set the destination port of the rule, use the following command:

```
destination port <NUMBER/WORD>
```

## Syntax Description

---

destination port	Set the destination port of the rule
<NUMBER/WORD>	Set the destination to this port number

---

### Defaults

Any.

### Example

```
ruckus(config-guest-restrict-access)# destination port 562
```

The command was executed successfully.

## protocol

To set the protocol for the rule, use the following command:

```
protocol <NUMBER/WORD>
```

## Syntax Description

---

protocol	Set the protocol for the rule
<NUMBER/WORD>	Set to this protocol

---

### Defaults

Any.

### Example

```
ruckus(config-guest-restrict-access)# protocol 69
```

The command was executed successfully.

## IPv6 Guest Restrict Access Commands

Use the IPv6 guest restrict access commands to configure IPv6 restrict access rules. To run these commands, you must first enter the `config-ipv6-guest-restrict-access` context.



## no restrict-access-order-ipv6

To delete a restrict access order, use the following command:

```
no restrict-access-order-ipv6 <NUMBER>
```

### Syntax Description

---

no restrict-access-order-ipv6	Delete a restrict access order
-------------------------------	--------------------------------

---

<NUMBER>	Delete this order ID
----------	----------------------

---

### Defaults

None.

### Example

```
ruckus(config-guest-access)# no restrict-access-order-ipv6 2  
The IPv6 Restricted Subnet Access entry has been removed from the  
Guest Access.  
ruckus(config-guest-access)#
```

## restrict-access-order-ipv6

To create a new restrict access order or modify an existing restrict access order, use the following command:

```
restrict-access-order-ipv6 <NUMBER>
```

This command enters the config-ipv6-guest-restrict-access context. The following commands are available from within this context:

### Syntax Description

---

help	Shows available commands
history	Shows a list of previously run commands.
abort	Exits the config-guest-restrict-access context without saving changes.
end	Saves changes, and then exits the config-guest-restrict-access context.

---

exit	Saves changes, and then exits the config-guest-restrict-access context.
quit	Exits the config-guest-restrict-access context without saving changes.
order <NUMBER>	Sets the guest access rule order.
description <WORD>	Sets the guest access rule description.
type [allow   deny]	Sets the guest access rule type to allow or deny.
destination [address <IPv6-ADDR>   port <NUMBER/WORD>	Sets the destination address/port of a guest access rule.
protocol <NUMBER/WORD>	Sets the protocol of a guest access rule.
icmpv6-type	Sets the ICMPv6 type of a Guest Access rule.
show	Displays restricted subnet access settings.

### **Example**

```
ruckus(config-guest-access)# restrict-access-order-ipv6 2
ruckus(config-ipv6-guest-restrict-access)# type allow
The command was executed successfully.
ruckus(config-ipv6-guest-restrict-access)# show
    Description=
    Type= Allow
    Destination Address= Any
    Destination Port= Any
    Protocol= Any
    ICMPv6 Type= Any
ruckus(config-ipv6-guest-restrict-access)# end
The IPv6 Restricted Subnet Access entry has been added to the Guest
Access.
Your changes have been saved.
ruckus(config-guest-access)#
```

### **show**

To display guest access restriction settings, use the following command:

```
show
```

## Syntax Description

---

show	Display guest access restriction settings
------	---

---

### Example

```
ruckus(config-ipv6-guest-restrict-access)# show
      Description=
      Type= Allow
      Destination Address= Any
      Destination Port= Any
      Protocol= Any
      ICMPv6 Type= Any
ruckus(config-ipv6-guest-restrict-access)#
```

### order

To configure the guest access rule order, use the following command:

```
order <NUMBER>
```

## Syntax Description

---

order	Set the order of a guest access rule
<NUMBER>	Assign the rule this order

---

### Defaults

None.

### Example

```
ruckus(config-ipv6-guest-restrict-access)# order 3
The command was executed successfully.
```

### description

To set the description of a guest access rule, use the following command:

```
description <WORD>
```

## Syntax Description

description	Set the description of a guest access rule
<WORD>	Set this as description

## Defaults

None.

## Example

```
ruckus(config-ipv6-guest-restrict-access)# description guestd3
```

The command was executed successfully.

## type allow

To set the guest access rule type to 'allow', use the following command:

```
type allow
```

## Syntax Description

type	Set the guest access rule type
allow	Set the rule type to 'allow'

## Defaults

Deny.

## Example

```
ruckus(config-ipv6-guest-restrict-access)# type allow
```

The command was executed successfully.

## type deny

To set the guest access rule type to 'deny', use the following command:

```
type deny
```

## Syntax Description

type	Set the guest access rule type
deny	Set the rule type to 'deny'

## Defaults

Deny.

## Example

```
ruckus(config-ipv6-guest-restrict-access)# type deny
The command was executed successfully.
```

## destination address

To set the destination address of the rule, use the following command:

```
destination address <IP-ADDR/WORD>
```

## Syntax Description

destination address	Set the destination address of the rule
IP-ADDR/WORD	Set the destination to this IP address

## Defaults

None.

## Example

```
ruckus(config-ipv6-guest-restrict-access)# destination address
fe80::/64
The command was executed successfully.
ruckus(config-ipv6-guest-restrict-access)#
```

## destination port

To set the destination port of the rule, use the following command:

```
destination port <NUMBER/WORD>
```

## Syntax Description

destination port	Set the destination port of the rule
<NUMBER/WORD>	Set the destination to this port number

## Defaults

None.

## Example

```
ruckus(config-ipv6-guest-restrict-access)# destination port 562
The command was executed successfully.
```

## protocol

To set the protocol for the rule, use the following command:

```
protocol <NUMBER/WORD>
```

## Syntax Description

protocol	Set the protocol for the rule
<NUMBER/WORD>	Set to this protocol

## Defaults

None.

## Example

```
ruckus(config-ipv6-guest-restrict-access)# protocol 69
The command was executed successfully.
```

## icmpv6-type

To set the ICMPv6 type of a Guest Access rule, use the following command:

```
icmpv6-type [any | number <NUMBER>]
```

## Defaults

Any.

### **Example**

```
ruckus(config-ipv6-guest-restrict-access)# icmpv6-type any
The command was executed successfully.
ruckus(config-ipv6-guest-restrict-access)#
```

## **Configure Hotspot Commands**

Use the `hotspot` commands to configure the controller's hotspot settings. To run these commands, you must first enter the `config-hotspot` context.

### **hotspot**

To create a new hotspot or edit an existing entry and enter the `config-hotspot` context, use the following command:

```
hotspot <WORD>
```

### **Syntax Description**

<code>hotspot</code>	Create or edit a hotspot service
<code>&lt;WORD&gt;</code>	Name of hotspot service

### **Defaults**

None.

### **Example**

```
ruckus(config)# hotspot hotspot1
The Hotspot entry 'hotspot1' has been loaded. To save the Hotspot
entry, type end or exit.
ruckus(config-hotspot)#
```

### **no hotspot**

To delete a hotspot record from the list, use the following command:

```
no hotspot <WORD>
```

## ***Syntax Description***

---

hotspot	Create or edit a hotspot service
<WORD>	Name of hotspot service

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config)# hotspot hotspot1
```

The Hotspot entry 'hotspot1' has been loaded. To save the Hotspot entry, type end or exit.

```
ruckus(config-hotspot)#
```

### **abort**

To exit the config-hotspot context without saving changes, use the abort command.

```
abort
```

## ***Syntax Description***

---

abort	Exit the hotspot settings without saving changes
-------	--

---

### ***Defaults***

None.

### ***Example***

```
ruckus(config-hotspot)# abort
```

No changes have been saved.

```
ruckus(config)#
```

### **end**

To save changes, and then exit the config-hotspot context, use the following command:



```
end
```

### **Syntax Description**

---

end	Save changes, and then exit the context
-----	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# end
The login page url can't be empty.
ruckus(config-hotspot)# end
The Hotspot entry has saved successfully.
Your changes have been saved.
ruckus(config)#
```

### **exit**

To save changes, and then exit the `config-hotspot` context, use the following command:

```
exit
```

### **Syntax Description**

---

exit	Save changes, and then exit the context
------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# exit
The login page url can't be empty
ruckus(config-hotspot)# exit
The Hotspot entry has saved successfully.
Your changes have been saved.
```

## quit

To exit the `config-hotspot` context without saving changes, use the `quit` command.

```
quit
```

### **Syntax Description**

---

<code>quit</code>	Exit the hotspot settings without saving changes
-------------------	--

---

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# quit  
No changes have been saved.  
ruckus(config)#
```

## show

To display the current hotspot settings, use the following command:

```
show
```

### **Syntax Description**

---

<code>show</code>	Display the current hotspot settings
-------------------	--------------------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# show  
Hotspot:  
ID:  
1:  
Name= h1
```

```

Login Page Url= http://172.18.110.122
Start Page= redirect to the URL that the user intends to visit.
Session Timeout= Disabled
Idle Timeout= Enabled
Timeout= 60 Minutes
Authentication Server= Local Database
Accounting Server= Disabled
Location ID=
Location Name=
Walled Garden 1=
Walled Garden 2=
Walled Garden 3=
Walled Garden 4=
Walled Garden 5=
Rules:
Order= 1
Description= h1_order1
Type= Deny
Destination Address= 192.168.20.20/24
Destination Port= 920
Protocol= 58
    
```

## **name**

To set the hotspot name, use the following command

```
name <WORD>
```

### ***Syntax Description***

name	Set the hotspot name
<WORD>	Set to this name

### ***Defaults***

None.

### ***Example***

```
ruckus(config-hotspot)# name ruckus1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## smartclient

Use the following command to enable WISPr smart client support

```
smartclient [secure https] [secure http] [wispr-only
secure https] [wispr-only secure-http] [info]
```

### Syntax Description

smartclient	Enable WISPr smartclient support.
secure https	Enables WISPr smart client support with HTTPS security.
secure http	Enables WISPr smart client support with no security.
wispr-only secure https	Enables only WISPr smart client support with HTTPS security.
wispr-only secure http	Enables only WISPr smart client support with no security.
info	Sets the instruction to guide user to login by Smart Client application.

### Defaults

None.

### Example

```
ruckus(config-hotspot)# smartclient secure https
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-hotspot)#
```

## no smartclient

To disable WISPr Smart Client support, use the following command:

```
no smartclient
```

## login-page

To set the URL of the hotspot login, use the following command:

```
login-page [original|<WORD>]
```

### Syntax Description

login-page	Set the URL of the hotspot login
<WORD>	Set to this URL
original	Redirect to the URL that the user intends to visit

### Defaults

None.

### Example

```
ruckus(config-hotspot)# login-page http://ruckuswireless.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## start-page

To set the URL or page to which the user will be redirected after logging into the hotspot, use the following command:

```
start-page [original | url <WORD>]
```

### Syntax Description

start-page	Set the URL or page to which the user will be redirected after logging into the hotspot
original	Redirect user to the original page he or she intended to visit
url <WORD>	Redirect use to another page. Set the URL of the page in <WORD>.

### Defaults

original

**Example**

```
ruckus(config-hotspot)# start-page url
http://www.ruckuswireless.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

**no session-timeout**

To disable the session timeout for hotspot usage, use the following command:

```
no session-timeout
```

**Syntax Description**


---

no session-timeout	Disable the session timeout for hotspot usage
--------------------	---

---

**Defaults**

None.

**Example**

```
ruckus(config-hotspot)# no session-timeout
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

**session-timeout**

To enable and set the session timeout for hotspot usage, use the following command:

```
session-timeout <minutes>
```

**Syntax Description**


---

session-timeout	Disable the session timeout for hotspot usage
<minutes>	Set the session timeout to this value (in minutes)

---

**Defaults**

1440 minutes

**Example**

```
ruckus(config-hotspot)# session-timeout 20
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

**no grace-period**

To disable the grace period (idle timeout) for hotspot users, use the following command:

```
no grace-period
```

**Syntax Description**


---

no grace-period	Disable the idle timeout for hotspot users
-----------------	--

---

**Defaults**

None.

**Example**

```
ruckus(config-hotspot)# no grace-period
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

**grace-period**

To enable and set the grace period (idle timeout) for hotspot users, use the following command:

```
grace-period <minutes>
```

**Syntax Description**


---

grace-period	Set the idle timeout for hotspot users
<minutes>	Set the idle timeout to this value (in minutes)

---

**Defaults**

60 minutes

### Example

```
ruckus(config-hotspot)# grace-period 20
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### auth-server local

To use ZoneDirector as the authentication server for hotspot users, use the following command:

```
auth-server local
```

### Syntax Description

auth-server	Set an authentication server for hotspot users
local	Use ZoneDirector as the authentication server

### Defaults

local

### Example

```
ruckus(config-hotspot)# auth-server local
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### auth-server name

To use an external server for authenticating hotspot users, use the following command:

```
auth-server name <WORD>
```

### Syntax Description

auth-server name	Set an external authentication server for hotspot users
<WORD>	Use this server as the authentication server

### Defaults



None.

### **Example**

```
ruckus(config-hotspot)# auth-server name radius1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-hotspot)#
```

### **auth-server name no-mac-bypass**

To disable MAC authentication bypass (no redirection), use the following command:

```
auth-server name <WORD> no-mac-bypass
```

### **auth-server name mac-bypass**

To enable MAC authentication bypass (no redirection) and use password as authentication password, use the following command:

```
auth-server name <WORD> mac-bypass [mac | password <WORD>]
```

### **Syntax Description**

auth-server name	Set an external authentication server for hotspot users
<WORD>	Authentication server name
mac-bypass	Enable MAC auth bypass
mac	Enables MAC authentication bypass (no redirection) and use device MAC address as authentication password.
password <WORD>	Enables MAC authentication bypass (no redirection) and use password as authentication password.
mac-in-dot1x	Use device MAC address as authentication password and enable to send username and password in 802.1X format of 00-10-A4-23-19-C0 (by default 0010a42319c0).

---

password-in-dot1x <WORD>	Use password as authentication password and enable to send username and password in 802.1Xformatof00-10-A4-23-19-C0(bydefault 0010a42319c0).
--------------------------	--

---

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# auth-server name radius1 mac-bypass mac
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hotspot)#
```

## **auth-server name mac-bypass mac-addr-format**

To set MAC auth username and password to one of the following formats, use the following command:

```
auth-server name <WORD> mac-bypass mac-addr-format
[FORMAT]
```

### **Syntax Description**

---

auth-server name	Set an external authentication server for hotspot users
<WORD>	Authentication server name
mac-bypass	Enable MAC auth bypass
mac-addr-format	Enable MAC authentication bypass (no redirection) and use device MAC address as authentication password.
[FORMAT]	Set the MAC address format.
aabbccddeeff	Set the MAC address format to aabbccddeeff.
aa-bb-cc-dd-ee-ff	Set the MAC address format to aa-bb-cc-dd-ee-ff.

---

aa:bb:cc:dd:ee:ff	Set the MAC address format to aa:bb:cc:dd:ee:ff.
AABBCCDDEEFF	Set the MAC address format to AABBCCDDEEFF.
AA-BB-CC-DD-EE-FF	Set the MAC address format to AA-BB-CC-DD-EE-FF.
AA:BB:CC:DD:EE:FF	Set the MAC address format to AA:BB:CC:DD:EE:FF.

### acct-server

To enable the accounting server for hotspot usage, use the following command:

```
acct-server <WORD>
```

### Syntax Description

acct-server	Enable the accounting server for hotspot usage
<WORD>	Name of the AAA server

### Defaults

None.

### Example

```
ruckus(config-hotspot)# acct-server "RADIUS Accounting"
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hotspot)#
```

### no acct-server

To disable the accounting server for hotspot usage, use the following command:

```
no acct-server
```

## Syntax Description

---

no acct-server	Disable the accounting server for hotspot usage
----------------	---

---

### Defaults

None.

### Example

```
ruckus(config-hotspot)# no acct-server
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## acct-server interim-update

To enable and set the accounting server for hotspot usage, use the following command:

```
acct-server <WORD> interim-update <NUMBER>
```

## Syntax Description

---

no acct-server	Enable and set the accounting server for hotspot usage
<WORD>	Set to this accounting server
interim-update	Set the interim update interval
<NUMBER>	Set to this interval (in minutes)

---

### Defaults

5 minutes

### Example

```
ruckus(config-hotspot)# acct-server asd interim-update 10
```

The AAA server 'asd' could not be found. Please check the spelling, and then try again.

```
ruckus(config-hotspot)# acct-server acct1 interim-update 20
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## client-isolation

To enable wireless client isolation (on AP or across APs), use the following command:

```
client-isolation [isolation-on-ap|isolation-across-ap]
                 [enable|disable]
```

### *Syntax Description*

client-isolation	Enable client isolation.
isolation-on-ap	Enable client isolation per AP.
isolation-on-subnet	Enable spoof guarding and across AP client isolation using whitelist.

### *Defaults*

Disabled

### *Example*

```
ruckus(config-hotspot)# client-isolation isolation-on-ap enable
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hotspot)# client-isolation isolation-on-subnet
enable
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hotspot)#
```

## whitelist

To apply a client isolation whitelist to this Hotspot, use the following command:

```
whitelist name <WORD>
```

## location-id

To set the location ID of the hotspot, use the following command:

```
location-id <location-id>
```

## Syntax Description

location-id	Set the location ID of the hotspot
<location-id>	Set to this location ID

### Defaults

None.

### Example

```
ruckus(config-hotspot)# location-id us
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## location-name

To set the location name of the hotspot, use the following command:

```
location-name <location-name>
```

## Syntax Description

location-name	Set the location name of the hotspot
<location-name>	Set to this location name

### Defaults

None.

### Example

```
ruckus(config-hotspot)# location-name shenzhen
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## walled-garden

To set a hotspot “walled garden” URL, use the following command:

```
walled-garden <INDEX> <WORD>
```

## Syntax Description

walled-garden	Create a walled garden rule
<INDEX>	Enter walled garden URL index. (1~35)
<WORD>	Destination URL

## Defaults

None.

## Example

```
ruckus(config-hotspot)# walled-garden 1 www.ruckuswireless.com
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hotspot)#
```

## no walled-garden

To delete a walled garden URL, use the following command

```
no walled-garden <INDEX>
```

## Syntax Description

walled-garden	Delete a walled garden rule
<INDEX>	Enter walled garden URL index. (1~35)

## Defaults

None.

## Example

```
ruckus(config-hotspot)# no walled-garden 1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hotspot)#
```

## Configuring Hotspot Restricted Access Rules

The following commands are used to create and modify Hotspot restricted access rules. Use the `restrict-access-order` command from the `config-hotspot` context to enter the `config-hotspot-restrict-access` context.

### **restrict-access-order**

To create a new restrict access order or modify an existing restrict access order, use the following command:

```
restrict-access-order <NUMBER>
```

### **Syntax Description**

<code>restrict-access-order</code>	Add a restrict access order
<code>&lt;NUMBER&gt;</code>	Add this order ID
<code>order &lt;NUMBER&gt;</code>	Sets the hotspot rule order.
<code>description &lt;WORD&gt;</code>	Sets the hotspot rule description.
<code>type allow</code>	Sets the hotspot rule type to 'allow'.
<code>type deny</code>	Sets the hotspot rule type to 'deny'.
<code>destination address &lt;IP-ADDR/WORD&gt;</code>	Sets the destination address of a hotspot rule.
<code>destination port &lt;NUMBER/WORD&gt;</code>	Sets the destination port of a hotspot rule.
<code>protocol &lt;NUMBER/WORD&gt;</code>	Sets the protocol of a hotspot rule.
<code>show</code>	Displays the policy rule.

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# restrict-access-order 1
ruckus(config-hotspot-restrict-access)#
ruckus(config-hotspot-restrict-access)# show
      Description=
      Type= Deny
```



```

Destination Address= Any
Destination Port= Any
Protocol= Any
ruckus(config-hotspot-restrict-access) #

```

## no restrict-access-order

To delete a restrict access order, use the following command:

```
no restrict-access-order <NUMBER>
```

### Syntax Description

<code>no restrict-access-order</code>	Delete a restrict access order
<code>&lt;NUMBER&gt;</code>	Delete this order ID

### Defaults

None.

### Example

```
ruckus(config-hotspot) # no restrict-access-order 1
```

The rule '1' has been removed from the Hotspot.

## restrict-access-order-ipv6

To create a new IPv6 restrict access order or modify an existing restrict access order, use the following command:

```
restrict-access-order-ipv6 <NUMBER>
```

### Syntax Description

<code>restrict-access-order-ipv6</code>	Add a restrict access order
<code>&lt;NUMBER&gt;</code>	Add this order ID
<code>order &lt;NUMBER&gt;</code>	Sets the hotspot rule order.
<code>description &lt;WORD&gt;</code>	Sets the hotspot rule description.

type allow	Sets the hotspot rule type to 'allow'.
type deny	Sets the hotspot rule type to 'deny'.
destination address <IP-ADDR/ WORD>	Sets the destination address of a hotspot rule.
destination port <NUMBER/ WORD>	Sets the destination port of a hotspot rule.
protocol <NUMBER/WORD>	Sets the protocol of a hotspot rule.
icmpv6 type [any number <NUMBER>]	Sets the icmpv6 type of a hotspot rule.
show	Displays the policy rule.

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# restrict-access-order-ipv6 1
ruckus(config-hotspot-restrict-access)#
ruckus(config-hotspot-restrict-access-ipv6)# show
    Description=
    Type= Deny
    Destination Address= Any
    Destination Port= Any
    Protocol= Any
    ICMPv6 Type= Any
ruckus(config-hotspot-restrict-access-ipv6)#
```

### **no restrict-access-order-ipv6**

To delete a restrict access order, use the following command:

```
no restrict-access-order-ipv6 <order_id>
```

### **Syntax Description**

no restrict-access-order	Delete a restrict access order
--------------------------	--------------------------------

---

<order_id>	Delete this order ID
------------	----------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(config-hotspot)# no restrict-access-order-ipv6 1
```

The rule '1' has been removed from the Hotspot.

### **icmpv6-type**

To set the ICMPv6 type, use the following command:

```
icmpv6-type [any | number <NUMBER>]
```

### **Defaults**

Any.

### **Example**

```
ruckus(config-hotspot-restrict-access-ipv6)# icmpv6-type any
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-hotspot-restrict-access-ipv6)#
```

## **Hotspot Access Restriction Commands**

Use the `hotspot-restrict-access` commands to configure network segments to which hotspot access will be blocked. To run these commands, you must first enter the `config-hotspot-restrict-access` context.

The same commands are available for IPv6 networks from the `config-hotspot-restrict-access-ipv6` context.

### **end**

To save changes, and then exit the `config-hotspot-restrict-access` context, use the following command:

```
end
```

## ***Syntax Description***

---

end	Save changes, and then exit the context
-----	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus (config-hotspot-restrict-access) # end  
ruckus (config-hotspot) #
```

### **exit**

To save changes, and then exit the `config-hotspot-restrict-access` context, use the following command:

```
exit
```

## ***Syntax Description***

---

exit	Save changes, and then exit the context
------	---

---

### ***Defaults***

None.

### ***Example***

```
ruckus (config-hotspot-restrict-access) # exit  
ruckus (config-hotspot) #
```

### **show**

To display hotspot access restriction settings, use the following command:

```
show
```

## ***Syntax Description***

---

show	Display the hotspot access restriction settings
------	---

---

## Defaults

None.

## order

To configure the hotspot access rule order, use the following command:

```
order <NUMBER>
```

## Syntax Description

order	Set the order of a hotspot access rule
<NUMBER>	Assign the rule this order

## Defaults

None.

## Example

```
ruckus(config-hotspot-restrict-access)# order 1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## description

To set the description of a hotspot access rule, use the following command:

```
description <WORD>
```

## Syntax Description

description	Set the description of a hotspot access rule
<WORD>	Set this as description

## Defaults

None.

## Example

```
ruckus(config-hotspot-restrict-access)# description h1_order1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## type allow

To set the hotspot access rule type to 'allow', use the following command:

```
type allow
```

### Syntax Description

type	Set the hotspot access rule type
allow	Set the rule type to 'allow'

### Defaults

None.

### Example

```
ruckus(config-hotspot-restrict-access)# type allow
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## type deny

To set the hotspot access rule type to 'deny', use the following command:

```
type deny
```

### Syntax Description

type	Set the hotspot access rule type
deny	Set the rule type to 'deny'

### Defaults

None.

### Example

```
ruckus(config-hotspot-restrict-access)# type deny
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## destination address

To set the destination address of the rule, use the following command:

```
destination address <IP-ADDR/WORD>
```

### Syntax Description

destination address	Set the destination address of the rule
IP-ADDR/WORD	Set the destination to this IP address

### Defaults

None.

### Example

```
ruckus(config-hotspot-restrict-access)# destination address
192.168.20.20/24
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## destination port

To set the destination port of the rule, use the following command:

```
destination port <NUMBER/WORD>
```

### Syntax Description

destination port	Set the destination port of the rule
<NUMBER/WORD>	Set the destination to this port number

### Defaults

None.

### Example

```
ruckus(config-hotspot-restrict-access)# destination port 920
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## protocol

To set the protocol for the rule, use the following command:

```
protocol <NUMBER/WORD>
```

### Syntax Description

---

protocol	Set the protocol for the rule
<NUMBER/WORD>	Set to this protocol

---

### Defaults

None.

### Example

```
ruckus(config-hotspot-restrict-access)# protocol 58
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## intrusion-prevention

To enable temporary blocking of Hotspot clients with repeated authentication attempts, use the following command:

```
intrusion-prevention
```

### Defaults

Disabled.

### Example

```
ruckus(config-hotspot)# intrusion-prevention
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-hotspot)#
```



## no intrusion-prevention

To disable temporary blocking of Hotspot clients with repeated authentication failure, use the following command:

```
no intrusion-prevention
```

### Example

```
ruckus(config-hotspot)# no intrusion-prevention
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-hotspot)#
```

## Configure Hotspot 2.0 Commands

Use the `hs20op` and `hs20sp` commands to configure the controller's Hotspot 2.0 operator and service provider settings. To run these commands, you must first enter the `config-hs20op` or `config-hs20sp` context.

To deploy a Hotspot 2.0 service, you must configure the following:

- A Hotspot 2.0 Operator entry
- A Hotspot 2.0 Service Provider entry
- A WLAN with Hotspot 2.0 service enabled

### hs20op

Use the following command to configure a Hotspot 2.0 Operator entry:

```
hs20op <WORD>
```

### Syntax Description

hs20op	Create or configure a Hotspot 2.0 Operator entry
<WORD>	The name of the Hotspot 2.0 Operator entry.

### Example

```
ruckus(config)# hs20op operator1
```

The Hotspot (2.0) operator entry 'operator1' has been created.

```
ruckus(config-hs20op)# end
```

The Hotspot (2.0) operator entry has saved successfully.

Your changes have been saved.

```
ruckus(config)#
```

## no hs20op

Use the following command to delete a Hotspot 2.0 Operator entry:

```
no hs20op <WORD>
```

### Example

```
ruckus(config)# no hs20op operator1
```

The Hotspot (2.0) operator 'operator1' has been deleted.

```
ruckus(config)#
```

## Configure Hotspot 2.0 Operator Settings

The following commands can be used to configure Hotspot 2.0 Operator entry settings. To execute these commands, you must first create or edit a Hotspot 2.0 Operator entry using the `hs20op` command and entering the `config-hs20op` context.

### Syntax Description

help	Shows available commands.
history	Shows a list of previously run commands.
abort	Exits the config-hs20op context without saving changes.
end	Saves changes, and then exits the config-hs20op context.
exit	Saves changes, and then exits the config-hs20op context.
quit	Exits the config-hs20op context without saving changes.
no internet-option	Disables with connectivity to internet.
no hessid	Sets the HESSID to empty.
no service-provider <WORD>	Deletes a service provider from the Hotspot (2.0) operator.
no venue-group-type	Sets both venue group and venue type to unspecified.

no friendly-name <LANGUAGE>	Disable the friendly name for the specified language.
no asra	Disables additional step required for access.
no asra terms	Disables ASRA Type: Acceptance of terms and conditions.
no asra enrollment	Disables ASRA Type: On-line enrollment supported.
no asra http-https	Disables ASRA Type: http/https redirection.
no asra dns	Disables ASRA Type: DNS redirection.
no asra http-https-url	Sets the redirect URL of http/https redirection to empty.
no wan-metrics sym	Disables Symmetric Link.
no wan-metrics at-cap	Disables WAN at Capability.
no custm-conn-cap <NUMBER>	Deletes a Connection Capability entry.
no adv-gas dos-detect	Disables the GAS DOS detection.
no hs-caps operating- class-indication	Disables the operating class indication.
name <WORD>	Sets the hotspot(2.0) operator entry name.
description <WORD>	Sets the hotspot(2.0) operator entry description.
internet-option	Enables with connectivity to internet.
hessid <MAC>	Sets the HESSID.
hessid-use-bssid	Sets the HESSID to use BSSID.
service-provider <WORD>	Adds a service provider to the Hotspot (2.0) operator.
venue-group-type unspecified	Sets the venue group to unspecified
venue-group-type assembly	Sets the venue group to assembly
venue-group-type assembly unspecified	Sets the venue type to unspecified
venue-group-type assembly arena	Sets the venue type to arena

venue-group-type assembly stadium	Sets the venue type to stadium
venue-group-type assembly passenger- terminal	Sets the venue type to passenger terminal
venue-group-type assembly amphitheater	Sets the venue type to amphitheater
venue-group-type assembly amusement- park	Sets the venue type to amusement park
venue-group-type assembly place-worship	Sets the venue type to place of worship
venue-group-type assembly convention- center	Sets the venue type to convention center
venue-group-type assembly library	Sets the venue type to library
venue-group-type assembly museum	Sets the venue type to museum
venue-group-type assembly restaurant	Sets the venue type to restaurant
venue-group-type assembly theater	Sets the venue type to theater
venue-group-type assembly bar	Sets the venue type to bar
venue-group-type assembly coffee-shop	Sets the venue type to coffee shop
venue-group-type assembly zoo-or- aquarium	Sets the venue type to zoo or aquarium
venue-group-type assembly emergency- coordination-center	Sets the venue type to emergency coordination center
venue-group-type business	Sets the venue group to business

---

venue-group-type business unspecified	Sets the venue type to unspecified
venue-group-type business doctor-or- dentist-office	Sets the venue type to doctor or dentist office
venue-group-type business bank	Sets the venue type to bank
venue-group-type business fire-station	Sets the venue type to fire station
venue-group-type business police-station	Sets the venue type to police station
venue-group-type business post-office	Sets the venue type to post office
venue-group-type business professional- office	Sets the venue type to professional office
venue-group-type business research-and- development-facility	Sets the venue type to research and development facility
venue-group-type business attorney-office	Sets the venue type to attorney office
venue-group-type educational	Sets the venue group to educational
venue-group-type educational unspecified	Sets the venue type to unspecified
venue-group-type educational school- primary	Sets the venue type to school primary
venue-group-type educational school- secondary	Sets the venue type to school secondary
venue-group-type educational university- or-college	Sets the venue type to university or college

---

venue-group-type factory-industrial	Sets the venue group to factory industrial
venue-group-type factory-industrial unspecified	Sets the venue type to unspecified
venue-group-type factory-industrial factory	Sets the venue type to factory
venue-group-type institutional	Sets the venue group to institutional
venue-group-type institutional unspecified	Sets the venue type to unspecified
venue-group-type institutional hospital	Sets the venue type to hospital
venue-group-type institutional long-term- care-facility	Sets the venue type to long term care facility
venue-group-type institutional alcohol- and-drug-rehabilitation- center	Sets the venue type to alcohol and drug reHabilitation center
venue-group-type institutional group-home	Sets the venue type to group home
venue-group-type institutional prison-or-jail	Sets the venue type to prison or jail
venue-group-type mercantile	Sets the venue group to mercantile
venue-group-type mercantile unspecified	Sets the venue type to unspecified
venue-group-type mercantile retail-store	Sets the venue type to retail store
venue-group-type mercantile grocery- market	Sets the venue type to grocery market

---

venue-group-type mercantile automotive- service-station	Sets the venue type to automotive service station
venue-group-type mercantile shopping- mall	Sets the venue type to shopping mall
venue-group-type mercantile gas-station	Sets the venue type to gas station
venue-group-type residential	Sets the venue group to residential
venue-group-type residential unspecified	Sets the venue type to unspecified
venue-group-type residential private- residence	Sets the venue type to private residence
venue-group-type residential hotel-or- motel	Sets the venue type to hotel or motel
venue-group-type residential dormitory	Sets the venue type to dormitory
venue-group-type residential boarding- house	Sets the venue type to boarding house
venue-group-type storage	Sets the venue group to storage
venue-group-type storage unspecified	Sets the venue type to unspecified
venue-group-type utility-miscellaneous	Sets the venue group to utility miscellaneous
venue-group-type utility-miscellaneous unspecified	Sets the venue type to unspecified
venue-group-type vehicular	Sets the venue group to vehicular

---

venue-group-type vehicular unspecified	Sets the venue type to unspecified
venue-group-type vehicular automobile-or-truck	Sets the venue type to automobile or truck
venue-group-type vehicular airplane	Sets the venue type to airplane
venue-group-type vehicular bus	Sets the venue type to bus
venue-group-type vehicular ferry	Sets the venue type to ferry
venue-group-type vehicular ship-or-boat	Sets the venue type to ship or boat
venue-group-type vehicular train	Sets the venue type to train
venue-group-type vehicular motor-bike	Sets the venue type to motor bike
venue-group-type outdoor	Sets the venue group to outdoor
venue-group-type outdoor unspecified	Sets the venue type to unspecified
venue-group-type outdoor muni-mesh-network	Sets the venue type to muni mesh network
venue-group-type outdoor city-park	Sets the venue type to city park
venue-group-type outdoor rest-area	Sets the venue type to rest area
venue-group-type outdoor traffic-control	Sets the venue type to traffic control
venue-group-type outdoor bus-stop	Sets the venue type to bus stop
venue-group-type outdoor kiosk	Sets the venue type to kiosk



friendly-name <LANGUAGE> <WORD>	Sets the friendly name for the specified language.
asra	Enables additional step required for access.
asra terms	Enables ASRA Type: Acceptance of terms and conditions.
asra enrollment	Enables ASRA Type: On-line enrollment supported.
asra http-https	Enables ASRA Type: http/https redirection.
asra http-https url <WORD>	Sets the redirect URL of http/https redirection.
asra dns	Enables ASRA Type: DNS redirection.
accs-net-type private	Sets the access network type to Private network.
accs-net-type private- with-guest	Sets the access network type to Private network with guest access.
accs-net-type chargeable-public	Sets the access network type to Chargeable public network.
accs-net-type free- public	Sets the access network type to Free public network.
accs-net-type personal- device	Sets the access network type to Personal device network.
accs-net-type test-or- experimental	Sets the access network type to Test or experimental.
accs-net-type wildcard	Sets the access network type to Wildcard.
ip-addr-type ipv4 not- avail	Sets the IPv4 Address Type to not available.
ip-addr-type ipv4 public	Sets the IPv4 Address Type to public address.
ip-addr-type ipv4 port- restricted	Sets the IPv4 Address Type to port-restricted address.
ip-addr-type ipv4 single- nated	Sets the IPv4 Address Type to single NATed private address.
ip-addr-type ipv4 double-nated	Sets the IPv4 Address Type to double NATed private address.
ip-addr-type ipv4 port- single	Sets the IPv4 Address Type to port-restricted address and single NATed private address.

ip-addr-type ipv4 port-double	Sets the IPv4 Address Type to port-restricted address and double NATed private address.
ip-addr-type ipv4 unknown	Sets the IPv4 Address Type to unknown.
ip-addr-type ipv6 not-avail	Sets the IPv6 Address Type to not available.
ip-addr-type ipv6 avail	Sets the IPv6 Address Type to available.
ip-addr-type ipv6 unknown	Sets the IPv6 Address Type to unknown.
wan-metrics sym	Enables Symmetric Link.
wan-metrics at-cap	Enables WAN at Capability.
wan-metrics link-stat up	Sets Link Status to Link UP.
wan-metrics link-stat down	Sets Link Status to Link Down.
wan-metrics link-stat test	Sets Link Status to Link in Test State.
wan-metrics downlink-load <NUMBER>	Sets WAN downlink load.
wan-metrics downlink-speed <NUMBER>	Sets WAN downlink speed.
wan-metrics uplink-load <NUMBER>	Sets WAN uplink load.
wan-metrics uplink-speed <NUMBER>	Sets WAN uplink speed.
wan-metrics lmd <NUMBER>	Sets Load Measurement Duration.
conn-cap icmp closed	Sets the ICMP Connection Capability Status to closed
conn-cap icmp open	Sets the ICMP Connection Capability Status to open
conn-cap icmp unknown	Sets the ICMP Connection Capability Status to unknown
conn-cap ftp closed	Sets the FTP Connection Capability Status to closed
conn-cap ftp open	Sets the FTP Connection Capability Status to open
conn-cap ftp unknown	Sets the FTP Connection Capability Status to unknown

conn-cap ssh closed	Sets the SSH Connection Capability Status to cloed
conn-cap ssh open	Sets the SSH Connection Capability Status to open
conn-cap ssh unknown	Sets the SSH Connection Capability Status to unknown
conn-cap http closed	Sets the HTTP Connection Capability Status to cloed
conn-cap http open	Sets the HTTP Connection Capability Status to open
conn-cap http unknown	Sets the HTTP Connection Capability Status to unknown
conn-cap tls-vpn closed	Sets the TLS VPN Connection Capability Status to cloed
conn-cap tls-vpn open	Sets the TLS VPN Connection Capability Status to open
conn-cap tls-vpn unknown	Sets the TLS VPN Connection Capability Status to unknown
conn-cap pptp-vpn closed	Sets the PPTP VPN Connection Capability Status to cloed
conn-cap pptp-vpn open	Sets the PPTP VPN Connection Capability Status to open
conn-cap pptp-vpn unknown	Sets the PPTP VPN Connection Capability Status to unknown
conn-cap voip-tcp closed	Sets the VoIP(TCP) Connection Capability Status to closed
conn-cap voip-tcp open	Sets the VoIP(TCP) Connection Capability Status to open
conn-cap voip-tcp unknown	Sets the VoIP(TCP) Connection Capability Status to unknown
conn-cap ikev2 closed	Sets the IKEv2 Connection Capability Status to cloed
conn-cap ikev2 open	Sets the IKEv2 Connection Capability Status to open
conn-cap ikev2 unknown	Sets the IKEv2 Connection Capability Status to unknown
conn-cap voip-udp closed	Sets the VoIP(UDP) Connection Capability Status to closed
conn-cap voip-udp open	Sets the VoIP(UDP) Connection Capability Status to open
conn-cap voip-udp unknown	Sets the VoIP(UDP) Connection Capability Status to unknown
conn-cap ipsec-vpn closed	Sets the IPSec VPN Connection Capability Status to cloed

conn-cap ipsec-vpn open	Sets the IPSec VPN Connection Capability Status to open
conn-cap ipsec-vpn unknown	Sets the IPSec VPN Connection Capability Status to unknown
conn-cap esp closed	Sets the ESP Connection Capability Status to closoed
conn-cap esp open	Sets the ESP Connection Capability Status to open
conn-cap esp unknown	Sets the ESP Connection Capability Status to unknown
custm-conn-cap <NUMBER> ip-proto <NUMBER> port <NUMBER> status closed	Sets Status to closed.
custm-conn-cap <NUMBER> ip-proto <NUMBER> port <NUMBER> status closed description <WORD>	Sets the description of Connection Capability entry.
custm-conn-cap <NUMBER> ip-proto <NUMBER> port <NUMBER> status open	Sets Status to open.
custm-conn-cap <NUMBER> ip-proto <NUMBER> port <NUMBER> status open description <WORD>	Sets the description of Connection Capability entry.
custm-conn-cap <NUMBER> ip-proto <NUMBER> port <NUMBER> status unknown	Sets Status to unknown.

custm-conn-cap <NUMBER> ip-PROTO <NUMBER> port <NUMBER> status unknown description <WORD>	Sets the description of Connection Capability entry.
adv-gas cb-delay <NUMBER>	Sets the GAS Comeback Delay.
adv-gas rsp-limit <NUMBER>	Sets the GAS query response length limit.
adv-gas rsp-buf-time <NUMBER>	Sets the GAS query response buffering time.
adv-gas dos-detect	Enables the GAS DOS detection.
adv-gas dos-maxreq <NUMBER>	Set the GAS DOS detection maximum request number.
hs-caps operating- class-indication 2.4	Sets the operating class indication to 2.4 GHz.
hs-caps operating- class-indication 5	Sets the operating class indication to 5 GHz.
hs-caps operating- class-indication dual- band	Sets the operating class indication to 2.4/5 GHz.
show	Displays hotspot 2.0 operator settings.

## hs20sp

Use the following command to configure a Hotspot 2.0 Service Provider entry:

```
hs20sp <WORD>
```

### Example

```
ruckus(config)# hs20sp serviceprovider1
```

The Hotspot (2.0) service provider entry 'serviceprovider1' has been created.

```
ruckus(config-hs20sp)# end
```

The Hotspot (2.0) service provider entry has saved successfully.

Your changes have been saved.

```
ruckus(config)#
```

## no hs20sp

Use the following command to delete a Hotspot 2.0 Service Provider entry:

```
no hs20sp <WORD>
```

### Example

```
ruckus(config)# no hs20sp provider1
```

The Hotspot (2.0) service provider 'provider1' has been deleted.

```
ruckus(config)#
```

## Configure Hotspot 2.0 Service Provider Settings

The following commands can be used to configure Hotspot 2.0 Service Provider entry settings. To execute these commands, you must first create or edit a Hotspot 2.0 Service Provider entry using the `hs20sp` command and entering the `config-hs20sp` context.

### Syntax Description

help	Shows available commands.
history	Shows a list of previously run commands.
abort	Exits the config-hs20sp context without saving changes.
end	Saves changes, and then exits the config-hs20sp context.
exit	Saves changes, and then exits the config-hs20sp context.
quit	Exits the config-hs20sp context without saving changes.
no nai-realm <NUMBER>	Deletes a NAI Realm entry.
no domain-name <NUMBER>	Deletes a domain name entry.
no roam-consortium <NUMBER>	Deletes a roaming consortium entry.

no anqp-3gpp-info <NUMBER>	Deletes a 3GPP cellular network information entry.
name <WORD>	Sets the hotspot(2.0) service provider entry name.
description <WORD>	Sets the hotspot(2.0) service provider entry description.
nai-realm <NUMBER>	Creates a new NAI Realm entry or modifies an existing entry.
domain-name <NUMBER>	Creates a new domain name entry or modifies an existing entry.
domain-name <NUMBER> name <WORD>	Sets the domain name of a domain name entry.
roam-consortium <NUMBER>	Creates a new roaming consortium entry or modifies an existing entry.
roam-consortium <NUMBER> org-id <HEX>	Sets the organization ID of a roaming consortium entry.
roam-consortium <NUMBER> org-id <HEX> name <WORD>	Sets the name of a roaming consortium entry.
anqp-3gpp-info <NUMBER>	Creates a 3GPP cellular network information entry or modifies an existing entry list.
anqp-3gpp-info <NUMBER> mcc <NUMBER>	Sets the MCC of 3GPP cellular network information entry.
anqp-3gpp-info <NUMBER> mcc <NUMBER> mnc <NUMBER>	Sets the MNC of 3GPP cellular network information entry.
anqp-3gpp-info <NUMBER> mcc <NUMBER> mnc <NUMBER> name <WORD>	Sets the name of 3GPP cellular network information entry.
show	Displays hotspot 2.0 service provider settings.

## nai-realm

To create, a new NAI Realm entry or modifies an existing entry, use the following command:

```
nai-realm <NUMBER>
```

This command enters the `config-hs20sp-nai-realm` context. The following commands can be executed from within this context.

### Syntax Description

<code>name</code>	Sets the name of the NAI Realm entry.
<code>encoding</code>	Sets the encoding of the NAI Realm entry.
<code>eap-method &lt;NUMBER&gt;</code>	Sets the EAP method #X of the NAI Realm entry. (X:1~4)
<code>no</code>	Contains commands that can be executed from within the context.
<code>show</code>	Displays NAI Realm settings.

### Example

```
ruckus(config-hs20sp)# nai-realm 1
ruckus(config-hs20sp-nai-realm)# name realm1
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hs20sp-nai-realm)# show
    Name= realm1
    Encoding= RFC-4282
    EAP Method #1= N/A
    EAP Method #2= N/A
    EAP Method #3= N/A
    EAP Method #4= N/A
ruckus(config-hs20sp-nai-realm)# end
To save the changes, type 'end' or 'exit'.
ruckus(config-hs20sp)# end
The Hotspot (2.0) service provider entry has saved successfully.
Your changes have been saved.
ruckus(config)#
```

### name

Use the following command to set the name of the NAI Realm entry:

```
name <WORD>
```



## encoding

Use the following command to set the encoding of the NAI Realm entry:

```
encoding [rfc-4282 | utf-8]
```

## eap-method

Use the following command to set the EAP method of the NAI Realm entry:

```
eap-method <NUMBER>
```

## eap-method eap-mthd

Use the following command to set the EAP method of the NAI Realm entry:

```
eap-method <NUMBER> eap-mthd [N/A | <NAME>]
```

## Syntax Description

N/A	Sets the EAP method of the NAI Realm entry to N/A.
MD5-Challenge	Sets the EAP method of the NAI Realm entry to MD5-Challenge.
EAP-TLS	Sets the EAP method of the NAI Realm entry to EAP-TLS.
EAP-CISCO	Sets the EAP method of the NAI Realm entry to EAP-Cisco.
EAP-SIM	Sets the EAP method of the NAI Realm entry to EAP-SIM.
EAP-TTLS	Sets the EAP method of the NAI Realm entry to EAP-SIM.
PEAP	Sets the EAP method of the NAI Realm entry to PEAP.
MSCHAP-V2	Sets the EAP method of the NAI Realm entry to EAP-MSCHAP-V2.
EAP-AKA	Sets the EAP method of the NAI Realm entry to EAP-AKA.
EAP-AKA-Prime	Sets the EAP method of the NAI Realm entry to EAP-AKA'.

---

Reserved	Sets the EAP method of the NAI Realm entry to Reserved.
----------	---

---

### **Syntax Description**

```
ruckus(config-hs20sp-nai-realm) # eap-method 1 eap-mthd EAP-TLS
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-hs20sp-nai-realm) #
```

### **eap-method auth-info**

To set the Auth Info of the EAP method, use the following command:

```
eap-method <NUMBER> auth-info <NUMBER>
```

### **Syntax Description**

---

auth-id	Sets the auth info ID of the auth info.
---------	---

---

auth-id expanded-EAP-method	Sets the Auth Info of the EAP method to expanded-EAP-method.
-----------------------------	--

---

auth-id expanded-EAP-method vndr-id <NUMBER>	Sets the vendor ID of the auth info.
---	--------------------------------------

---

auth-id expanded-EAP-method vndr-id <NUMBER> vndr-type <NUMBER>	Sets the vendor type of the auth info.
---	--

---

auth-id nonEAP-inner-auth	Sets the Auth Info of the EAP method to Non-EAP Inner Authentication Type.
---------------------------	--

---

auth-id nonEAP-inner-auth auth- type	Sets the auth info type of the auth info.
---	---

---

auth-id nonEAP-inner-auth auth- type Reserved	Sets the Non-EAP Inner Authentication Type to Reserved.
--	---

---

auth-id nonEAP-inner-auth auth- type PAP	Sets the Non-EAP Inner Authentication Type to PAP.
---	--

---

auth-id nonEAP-inner-auth auth- type CHAP	Sets the Non-EAP Inner Authentication Type to CHAP.
--	---

---

auth-id nonEAP-inner-auth auth- type MSCHAP	Sets the Non-EAP Inner Authentication Type to MSCHAP.
--	---

---

auth-id nonEAP-inner-auth auth-type MSCHAPV2	Sets the Non-EAP Inner Authentication Type to MSCHAPV2.
auth-id inner-auth-EAP-mthd	Sets the Auth Info of the EAP method to Inner Authentication EAP Method Type.
auth-id inner-auth-EAP-mthd auth-type	Sets the auth info type of the auth info.
auth-id inner-auth-EAP-mthd auth-type EAP-TLS	Sets the Inner Authentication EAP Method Type to EAP-TLS.
auth-id inner-auth-EAP-mthd auth-type EAP-SIM	Sets the Inner Authentication EAP Method Type to EAP-SIM.
auth-id inner-auth-EAP-mthd auth-type EAP-TTLS	Sets the Inner Authentication EAP Method Type to EAP-TTLS.
auth-id inner-auth-EAP-mthd auth-type EAP-AKA	Sets the Inner Authentication EAP Method Type to EAP-AKA.
auth-id inner-auth-EAP-mthd auth-type EAP-AKA-Prime	Sets the Inner Authentication EAP Method Type to EAP-AKA'.
auth-id exp-inner-EAP-mthd	Sets the Auth Info of the EAP method to expanded-inner-EAP-method.
auth-id exp-inner-EAP-mthd vndr-id <NUMBER>	Sets the vendor ID of the auth info.
auth-id exp-inner-EAP-mthd vndr-id <NUMBER> vndr-type <NUMBER>	Sets the vendor type of the auth info.
auth-id credential-type	Sets the Auth Info of the EAP method to Credential Type.
auth-id credential-type auth-type	Sets the auth info type of the auth info.
auth-id credential-type auth-type SIM	Sets the Credential Type to SIM.
auth-id credential-type auth-type USIM	Sets the Credential Type to USIM.
auth-id credential-type auth-type NFC-secure-elem	Sets the Credential Type to NFC Secure Element.
auth-id credential-type auth-type hardware-token	Sets the Credential Type to Hardware Token.

auth-id credential-typeauth-type softoken	Sets the Credential Type to Softoken.
auth-id credential-typeauth-type certificate	Sets the Credential Type to Certificate.
auth-id credential-typeauth-type username-password	Sets the Credential Type to username/password.
auth-id credential-typeauth-type none	Sets the Credential Type to none.
auth-id credential-typeauth-type reserved	Sets the Credential Type to Reserved.
auth-id tunnel-EAP-mthd-crdn-type	Sets the Auth Info of the EAP method to Tunneled EAP Method Credential Type.
auth-id tunnel-EAP-mthd-crdn-type auth-type	Sets the auth info type of the auth info.
auth-id tunnel-EAP-mthd-crdn-type auth-type SIM	Sets the Tunneled EAP Method Credential Type to SIM.
auth-id tunnel-EAP-mthd-crdn-type auth-type USIM	Sets the Tunneled EAP Method Credential Type to USIM.
auth-id tunnel-EAP-mthd-crdn-type auth-type NFC-secure-elem	Sets the Tunneled EAP Method Credential Type to NFC Secure Element.
auth-id tunnel-EAP-mthd-crdn-type auth-type hardware-token	Sets the Tunneled EAP Method Credential Type to Hardware Token.
auth-id tunnel-EAP-mthd-crdn-type auth-type softoken	Sets the Tunneled EAP Method Credential Type to Softoken.
auth-id tunnel-EAP-mthd-crdn-type auth-type certificate	Sets the Tunneled EAP Method Credential Type to Certificate.
auth-id tunnel-EAP-mthd-crdn-type auth-type username-password	Sets the Tunneled EAP Method Credential Type to username/password.
auth-id tunnel-EAP-mthd-crdn-type auth-type reserved	Sets the Tunneled EAP Method Credential Type to Reserved.
auth-id tunnel-EAP-mthd-crdn-type auth-type anonymous	Sets the Tunneled EAP Method Credential Type to Anonymous.

<code>no eap-method &lt;NUMBER&gt;</code>	Sets the EAP method #X of the NAI Realm entry. (X:1~4)
<code>no eap-method &lt;NUMBER&gt;</code> <code>auth-info &lt;NUMBER&gt;</code>	Disable the Auth Info of the EAP method
<code>show</code>	Displays NAI Realm settings.

## Configure Mesh Commands

Use the `mesh` commands to configure the controller's mesh networking settings. To run these commands, you must first enter the `config-mesh` context.

### mesh

Use the `mesh` command to enter the `config-mesh` context and configure the mesh-related settings.

```
mesh
```

### Syntax Description

<code>mesh</code>	Configure mesh settings
-------------------	-------------------------

### Defaults

```
none
```

### Example

```
ruckus(config)# mesh
ruckus(config-mesh)#
```

### abort

To exit the `config-mesh` context without saving changes, use the `abort` command.

### end

To save changes, and then exit the `config-mesh` context, use the `end` command.

## exit

To save changes, and then exit the `config-mesh` context, use the `exit` command.

## quit

To exit the `config-mesh` context without saving changes, use the `quit` command.

## show

To display the current mesh settings, use the following command:

```
show
```

## Syntax Description

---

show	Display the current mesh settings
------	-----------------------------------

---

## Defaults

None.

## Example

```
ruckus(config-mesh)# show
Mesh Settings:
Mesh Status= Enabled
Mesh Name (ESSID)= Mesh-00000000311
Mesh Passphrase= GdxW5CUgNn_SEHOPyCSxv_chHScA MH-OpnRGfX sRvwXBJL-
wUsD64eK8CMEZfm
Mesh Hop Detection:
Status= Disabled
Mesh Downlinks Detection:
Status= Disabled
Tx. Rate of Management Frame=2Mbps
Beacon Interval= 200ms
ruckus(config-mesh)#
```

## ssid

To set the SSID of the mesh network, use the following command:

```
ssid <WORD/SSID>
```

### **Syntax Description**

ssid	Set the SSID of the mesh network
<WORD/SSID>	Set to this SSID

### **Defaults**

None.

### **Example**

```
ruckus(config-mesh)# ssid rks_mesh
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **passphrase**

To set the passphrase that allows access to the mesh network, use the following command:

```
passphrase <WORD>
```

### **Syntax Description**

passphrase	Set the passphrase that allows access to the mesh network
<WORD>	Set to this passphrase

### **Defaults**

None.

### **Example**

```
ruckus(config-mesh)# passphrase test123456
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## hops-warn-threshold

To enable and configure the mesh hop threshold, use the following command:

```
hops-warn-threshold <NUMBER>
```

### Syntax Description

---

hops-warn-threshold	Set the mesh hop threshold (max hops)
<NUMBER>	Set to this threshold value

---

### Defaults

5

### Example

```
ruckus(config-mesh)# hops-warn-threshold 6
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## no detect-hops

To disable the mesh hop threshold, use the following command:

```
no detect-hops
```

### Syntax Description

---

no detect-hops	Disable the mesh hop threshold
----------------	--------------------------------

---

### Defaults

None.

### Example

```
ruckus(config-mesh)# no detect-hops
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.



## fan-out-threshold

To enable and configure the mesh downlink threshold, use the following command:

```
fan-out-threshold <NUMBER>
```

### Syntax Description

fan-out-threshold	Set the mesh downlink threshold (max downlinks)
<NUMBER>	Set to this threshold value

### Defaults

5

### Example

```
ruckus(config-mesh)# fan-out-threshold 8
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## no detect-fanout

To disable the mesh downlink threshold, use the following command:

```
no detect-fanout
```

### Syntax Description

no detect-fanout	Disable the mesh downlink threshold
------------------	-------------------------------------

### Example

```
ruckus(config-mesh)# no detect-fanout
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## beacon-interval

To set the beacon interval for mesh links, use the following command:

```
beacon-interval <NUMBER>
```

## Syntax Description

beacon-interval	Set the beacon interval for mesh links
<NUMBER>	Enter the beacon interval (100~1000 TUs)

## Defaults

200

## Example

```
ruckus(config-mesh)# beacon-interval 200
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-mesh)#
```

## mgmt-tx-rate

To set the transmit rate for management frames, use the following command:

```
mgmt-tx-rate <RATE>
```

## Syntax Description

mgmt-tx-rate	Set the max transmit rate for management frames
<RATE>	Set the transmit rate (in Mbps).

## Defaults

2

## Example

```
ruckus(config-mesh)# mgmt-tx-rate 2
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-mesh)#
```

## mesh-uplink-selection static

Sets static on mesh uplinks, the default is static.

```
mesh-uplink selection static
```

### *Syntax Description*

---

mesh-uplink-selection	Set the mesh uplink selection method.
static	Set mesh uplink selection to static.

---

### *Defaults*

Static

### *Example*

```
ruckus(config-mesh)# mesh-uplink-selection static
Nothing changed
ruckus(config-mesh)#
```

## mesh-uplink-selection dynamic

Sets dynamic on mesh uplinks.

```
mesh-uplink selection dynamic
```

### *Syntax Description*

---

mesh-uplink-selection	Set the mesh uplink selection method.
dynamic	Set mesh uplink selection to dynamic.

---

### *Defaults*

Static

### *Example*

```
ruckus(config-mesh)# mesh-uplink-selection dynamic
The command was executed successfully. To save the changes, type
'end' or 'exit'.
```

```
ruckus(config-mesh)#
```

## Configure Alarm Commands

Use the `alarm` commands to configure the controller's alarm notification settings. To run these commands, you must first enter the `config-alarm` context.

### alarm

To enter the `config-alarm` context, use the following command.

```
alarm
```

### Example

```
ruckus(config)# alarm  
ruckus(config-alarm)#
```

### no alarm

To disable alarm settings, use the following command:

```
no alarm
```

### Example

```
ruckus(config)# no alarm  
The Alarm settings have been updated.  
ruckus(config)#
```

### abort

To exit the `config-alarm` context without saving changes, use the `abort` command.

```
abort
```

### Syntax Description

---

<code>abort</code>	Exit the alarm settings without saving changes
--------------------	--

---

### Defaults

None.

### **Example**

```
ruckus(config-alarm)# abort  
No changes have been saved.  
ruckus(config)#
```

### **end**

To save changes, and then exit the `config-alarm` context, use the following command:

```
end
```

### **Syntax Description**

---

<code>end</code>	Save changes, and then exit the context
------------------	---

---

### **Defaults**

None.

### **Example**

```
ruckus(config-alarm)# end  
The Alarm settings have been updated.  
Your changes have been saved.  
ruckus(config)#
```

### **exit**

To save changes, and then exit the `config-alarm` context, use the following command:

```
exit
```

### **Syntax Description**

---

<code>exit</code>	Save changes, and then exit the context
-------------------	---

---

## Defaults

None.

## Example

```
ruckus(config-alarm)# exit
```

The Alarm settings have been updated.

Your changes have been saved.

## quit

To exit the `config-alarm` context without saving changes, use the `quit` command.

```
quit
```

## Syntax Description

---

<code>quit</code>	Exit the alarm settings without saving changes
-------------------	--

---

## Defaults

None.

## Example

```
ruckus(config-alarm)# quit
```

No changes have been saved.

```
ruckus(config)#
```

## show

To display the current alarm settings, use the following command:

```
show
```

## Syntax Description

---

<code>show</code>	Display the current alarm settings
-------------------	------------------------------------

---

## Defaults

None.

### **Example**

```
ruckus(config)# alarm
ruckus(config-alarm)# show
Alarm:
  Status= Enabled
  Email Address= johndoe@gmail.com
  E-mail From = zonedirector@ruckuswireless.com
  SMTP Server Name= smtp.gmail.com
  SMTP Server Port= 587
  SMTP Authentication Username= johndoe@gmail.com
  SMTP Authentication Password= *****
  wait time=
  SMTP Encryption Options:
    TLS= Enabled
    STARTTLS= Enabled

ruckus(config-alarm)#
```

### **e-mail**

To set the email address to which alarm notifications will be sent, use the following command:

```
e-mail <WORD>
```

### **Syntax Description**

e-mail	Set the email address to which alarm notifications will be sent
<WORD>	Send alarm notifications to this email address

### **Defaults**

None.

### **Example**

```
ruckus(config-alarm)# e-mail joe@163.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## from

To set the sender from address for email alarms, use the following command:

```
from <WORD>
```

### Syntax Description

from	Set the email address from which alarm notifications will be sent
<WORD>	Send alarm notifications from this email address

### Defaults

None.

### Example

```
ruckus(config-alarm)# from zonedirector@zonedirector.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-alarm)#
```

## smtp-server-name

To set the SMTP server that ZoneDirector uses to send alarm notifications, use the following command:

```
smtp-server-name <WORD>
```

### Syntax Description

smtp-server-name	Set the SMTP server that ZoneDirector uses to send alarm notifications
<WORD>	Set to this SMTP server name



## Defaults

None.

## Example

```
ruckus(config-alarm)# smtp-server-name smtp.163.com
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## smtp-server-port

To set the SMTP server port that ZoneDirector uses to send alarm notifications, use the following command:

```
smtp-server-port <NUMBER>
```

## Syntax Description

smtp-server-port	Set the SMTP server port that ZoneDirector uses to send alarm notifications
<NUMBER>	Set to this SMTP server port

## Defaults

587

## Example

```
ruckus(config-alarm)# smtp-server-port 25
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## smtp-auth-name

To set the user name that ZoneDirector uses to authenticate with the SMTP server, use the following command:

```
smtp_auth_name <WORD>
```

## Syntax Description

smtp_auth_name	Set the user name that ZoneDirector uses to authenticate with the SMTP server
<WORD>	Set to this user name

## Defaults

None.

## Example

```
ruckus(config-alarm)# smtp-auth-name joe
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## smtp-auth-password

To set the password that ZoneDirector uses to authenticate with the SMTP server, use the following command:

```
smtp-auth-password <WORD>
```

## Syntax Description

smtp-auth-password	Set the password that ZoneDirector uses to authenticate with the SMTP server
<WORD>	Set to this password

## Defaults

None.

## Example

```
ruckus(config-alarm)# smtp-auth-password 123456
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

## smtp-wait-time

To set the SMTP server wait time (in seconds), use following command:

```
smtp-wait-time <NUMBER>
```

### **Example**

```
ruckus(config-alarm)# smtp-wait-time 10
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-alarm)#
```

### **tls-smtp-encryption**

To enable TLS for SMTP encryption of alarm notifications, use the following command:

```
tls-smtp-encryption [tls|starttls]
```

### **Syntax Description**

tls-smtp-encryption	Enable SMTP encryption of alarm notifications
tls	Enable TLS encryption for alarm notifications
starttls	Enable STARTTLS encryption for alarm notifications

### **Defaults**

None.

### **Example**

```
ruckus(config-alarm)# tls-smtp-encryption tls
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

### **no tls-smtp-encryption**

To disable TLS for SMTP encryption of alarm notifications, use the following command:

```
no tls-smtp-encryption [tls | starttls]
```

## Syntax Description

no tls-smtp-encryption	Disable SMTP encryption of alarm notifications
tls	Disable TLS encryption
starttls	Disable STARTTLS encryption

## Defaults

None.

## Example

```
ruckus(config-alarm)# no tls-smtp-encryption tls
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

# Configure Alarm-Event Settings

Use the alarm-event commands to configure which events will trigger ZoneDirector email alerts. Entering this command enters the `config-alarm-event` context.

## alarm-event

To enter the `config-alarm-event` context and configure email alarm notifications for specific event types, use the following command:

```
alarm-event
```

## event

To enable email alarm notifications for a specific alarm event, use the following command:

```
event <WORD>
```

## Syntax Description

event all	Enable email alarms for all event types
rogue-ap-detected	Enable email notification when Rogue AP detected

rogue-device-detected	Enable email notification when Ad hoc network detected
ap-lost-contacted	AP lost contact
ssid-spoofing-ap-detected	SSID spoofing AP detected
mac-spoofing-ap-detected	MAC spoofing AP detected
user-blocked-ap-detected	User blocked AP detected
rogue-dhcp-server-detected	Rogue DHCP server detected
temporary-license-expired	Temporary license has expired
temporary-license-will-expire	Temporary license will expire
lan-rogue-ap-detected	LAN Rogue AP detected
aaa-server-unreachable	AAA server unreachable
ap-has-hardware-problem	AP hardware problem detected
uplink-ap-lost	Mesh AP uplink connection lost
incomplete-primary/secondary-ip-settings	AP fails to maintain primary/secondary ZD IP address settings
smart-redundancy-state-changed	Smart Redundancy device status change detected
smart-redundancy-active-connected	Smart Redundancy device active device connected
smart-redundancy-standby-connected	Smart Redundancy standby device connected
smart-redundancy-active-disconnected	Smart Redundancy active device disconnected
smart-redundancy-standby-disconnected	Smart Redundancy standby device disconnected
entitlement-download-fail	Failure to download the Support Entitlement file from the Ruckus Entitlement server
test-alarm ap-lose-connection	Test AP connection lost alarm event
show	Show alarm settings

### **Defaults**

All enabled

**Example**

```

ruckus(config)# alarm-event
ruckus(config-alarm-event)# event all
ruckus(config-alarm-event)# show
Alarm Events Notify By Email:
  MSG_rogue_AP_detected=                enabled
  MSG_ad_hoc_network_detected=          enabled
  MSG_AP_lost=                          enabled
  MSG_SSID_spoofing_AP_detected=        enabled
  MSG_MAC_spoofing_AP_detected=         enabled
  MSG_admin_rogue_dhcp_server=          enabled
  MSG_admin_templc_expired=             enabled
  MSG_admin_templc_oneday=              enabled
  MSG_same_network_spoofing_AP_detected= enabled
  MSG_RADIUS_service_outage=           enabled
  MSG_AP_hardware_problem=              enabled
  MSG_AP_no_mesh_uplink=                enabled
  MSG_AP_keep_no_AC_cfg=                enabled
  MSG_cltr_change_to_active=            enabled
  MSG_cltr_active_connected=            enabled
  MSG_cltr_standby_connected=           enabled
  MSG_cltr_active_disconnected=         enabled
  MSG_cltr_standby_disconnected=        enabled
  MSG_user_blocked_AP_detected=         enabled
  MSG_Entitlement_file_download_fail=    enabled

ruckus(config-alarm-event)#

```

**no event**

To disable email alarm notifications for specific event types, use the following command:

```
no event <event_name>
```

**Syntax Description**


---

no event	Disable email alarms for this event type
----------	--

---

all	Disable email alarms for all event types
rogue-ap-detected	Rogue AP detected
rogue-device-detectedq	Ad hoc network detected
ap-lost-contacted	AP lost contact
ssid-spoofing-ap-detected	SSID spoofing AP detected
mac-spoofing-ap-detected	MAC spoofing AP detected
user-blocked-ap-detected	User blocked AP detected
rogue-dhcp-server-detected	Rogue DHCP server detected
temporary-license-expired	Temporary license has expired
temporary-license-will-expire	Temporary license will expire
lan-rogue-ap-detected	LAN Rogue AP detected
aaa-server-unreachable	AAA server unreachable
ap-has-hardware-problem	AP hardware problem detected
uplink-ap-lost	Mesh AP uplink connection lost
incomplete-primary/secondary-ip-settings	AP fails to maintain primary/secondary ZD IP address settings
smart-redundancy-state-changed	Smart Redundancy device status change detected
smart-redundancy-active-connected	Smart Redundancy device active device connected
smart-redundancy-standby-connected	Smart Redundancy standby device connected
smart-redundancy-active-disconnected	Smart Redundancy active device disconnected
smart-redundancy-standby-disconnected	Smart Redundancy standby device disconnected
entitlement-download-fail	Failure to download the Support Entitlement file from the Ruckus Entitlement server

### **Example**

```
ruckus(config-alarm-event) # no event aaa-server-unreachable
ruckus(config-alarm-event) # show
Alarm Events Notify By Email:
```

```
MSG_rogue_AP_detected=          enabled
MSG_ad_hoc_network_detected=    enabled
MSG_AP_lost=                    enabled
MSG_SSID_spoofing_AP_detected=  enabled
MSG_MAC_spoofing_AP_detected=   enabled
MSG_admin_rogue_dhcp_server=    enabled
MSG_admin_templc_expired=       enabled
MSG_admin_templc_oneday=        enabled
MSG_same_network_spoofing_AP_detected= enabled
MSG_RADIUS_service_outage=      disabled
MSG_AP_hardware_problem=        enabled
MSG_AP_no_mesh_uplink=          enabled
MSG_AP_keep_no_AC_cfg=          enabled
MSG_cltr_change_to_active=      enabled
MSG_cltr_active_connected=      enabled
MSG_cltr_standby_connected=     enabled
MSG_cltr_active_disconnected=   enabled
MSG_cltr_standby_disconnected=  enabled
MSG_user_blocked_AP_detected=   enabled
MSG_Entitlement_file_download_fail= enabled
```

```
ruckus(config-alarm-event)#
```

## Configure Services Commands

Use the `services` commands to configure miscellaneous service settings, such as automatic power and channel selection settings, ChannelFly, background scanning, rogue AP and rogue DHCP server detection, etc. To run these commands, you must first enter the `config-services` context.

### abort

To exit the `config-services` context without saving changes, use the `abort` command.

```
abort
```



## ***Syntax Description***

---

abort	Exit the service settings without saving changes
-------	--

---

### ***Example***

```
ruckus(config-services)# abort
No changes have been saved.
ruckus(config)#
```

### **end**

To save changes, and then exit the `config-services` context, use the following command:

```
end
```

## ***Syntax Description***

---

end	Save changes, and then exit the context
-----	---

---

### ***Example***

```
ruckus(config-services)# end
Your changes have been saved.
ruckus(config)#
```

### **exit**

To save changes, and then exit the `config-services` context, use the following command:

```
exit
```

## ***Syntax Description***

---

exit	Save changes, and then exit the context
------	---

---

### **Example**

```
ruckus(config-services)# exit
Your changes have been saved.
ruckus(config)#
```

### **quit**

To exit the `config-services` context without saving changes, use the `quit` command.

```
quit
```

### **Syntax Description**

---

<code>quit</code>	Exit the service settings without saving changes
-------------------	--

---

### **Example**

```
ruckus(config-services)# quit
No changes have been saved.
ruckus(config)#
```

### **auto-adjust-ap-power**

To enable the auto adjustment of the AP radio power, which helps optimize radio coverage when radio interference is present, use the following command:

```
auto-adjust-ap-power
```

### **Syntax Description**

---

<code>auto-adjust-ap-power</code>	Enable the auto adjustment of the AP radio power
-----------------------------------	--

---

### **Defaults**

Disabled.

### **Example**

```
ruckus(config-services)# auto-adjust-ap-power
The command was executed successfully.
```

## no auto-adjust-ap-power

To disable the auto adjustment of the AP radio power, which helps optimize radio coverage when radio interference is present, use the following command:

```
no auto-adjust-ap-power
```

### **Syntax Description**

---

no auto-adjust-ap-power	Disable the auto adjustment of the AP radio power
-------------------------	---

---

### **Defaults**

Disabled.

### **Example**

```
ruckus(config-services)# no auto-adjust-ap-power
```

The command was executed successfully.

## auto-adjust-ap-channel

To enable the auto adjustment of the AP radio channel when radio interference is present, use the following command:

```
auto-adjust-ap-channel
```

### **Syntax Description**

---

auto-adjust-ap-channel	Enable the auto adjustment of the AP radio channel
------------------------	--

---

### **Defaults**

None.

### **Example**

```
ruckus(config-services)# auto-adjust-ap-channel
```

The command was executed successfully.

## no auto-adjust-ap-channel

To disable the auto adjustment of theAP radio channel when radio interference is present, use the following command:

```
no auto-adjust-ap-channel
```

### *Syntax Description*

---

no auto-adjust-ap-channel	Disable the auto adjustment of theAP radio channel
---------------------------	--

---

### *Defaults*

None.

### *Example*

```
ruckus(config-services)# no auto-adjust-ap-channel
```

The command was executed successfully.

## raps

To enable the Radar Avoidance Pre-Scanning (RAPS) feature on supported access points (SC-8800-S, 7782, 7781, etc.), use the following command:

```
raps
```

## no raps

To disable the Radar Avoidance Pre-Scanning (RAPS) feature on supported access points (SC-8800-S, 7782, 7781, etc.), use the following command:

```
no raps
```

## channelfly

To enable ChannelFly channel management, use the following command:

```
channelfly [radio-2.4-mtbc | radio-5-mtbc] <NUMBER>
```

### *Syntax Description*

---

channelfly	Enable ChannelFly automatic adjustment of theAP radio channel
------------	---

---

radio-2.4	Enable ChannelFly on the 2.4 GHz radio
radio-5	Enable ChannelFly on the 5 GHz radio
mtbc	Set the mean time between channel changes
<NUMBER>	Number in minutes (1~1440) to set as mean time between channel change

### Defaults

Enabled for both 2.4 and 5 GHz radios

MTBC: 100

### Example

```
ruckus(config-services)# channelfly radio-2.4 100
The command was executed successfully.
ruckus(config-services)#
```

### Example

```
ruckus(config-services)# channelfly radio-2.4-mtbc 100
The command was executed successfully.
ruckus(config-services)#
```

### no channelfly

To disable ChannelFly channel management, use the following command:

```
no channelfly [radio-2.4 | radio-5]
```

### Syntax Description

no channelfly	Disable ChannelFly automatic adjustment of theAP radio channel
radio-2.4	Disable ChannelFly on the 2.4 GHz radio
radio-5	Disable ChannelFly on the 5 GHz radio

### Defaults

None.

### Example

```
ruckus(config-services)# no channelfly radio-2.4
The command was executed successfully.
ruckus(config-services)# no channelfly radio-5
The command was executed successfully.
ruckus(config-services)#
```

### background-scan

To enable background scanning and configure the scan interval, use the following command:

```
background-scan [radio-2.4-interval | radio-5-interval]
<NUMBER>
```

### Syntax Description

background-scan	Enable background scanning and configure the scan interval
radio-2.4-interval	Configure background scanning interval for the 2.4 GHz radio
radio-5-interval	Configure background scanning interval for the 5GHz radio
<NUMBER>	Perform background scan at this interval (in seconds)

### Defaults

20 seconds

### Example

```
ruckus(config-services)# background-scan radio-2.4-interval 6
The command was executed successfully.
```

### no background-scan

To disable background scanning on the 2.4GHz radio, use the following command:

```
no background-scan [radio-2.4|radio-5]
```

## Syntax Description

no background-scan	Disable background scanning
radio-2.4	Disable background scanning on the 2.4GHz radio
radio-5	Disable background scanning on the 5GHz radio

## Defaults

None

## Example

```
ruckus(config-services)# no background-scan radio-2.4
The command was executed successfully.
ruckus(config-services)# no background-scan radio-5
The command was executed successfully.
```

## aeroscout-detection

To enable detection of AeroScout RFID Tags by APs that are managed by ZoneDirector, use the following command:

```
aeroscout-detection
```

## Syntax Description

aeroscout-detection	Enable detection of AeroScout RFID Tags by APs
---------------------	--

## Defaults

Disabled

## Example

```
ruckus(config-services)# aeroscout-detection
The command was executed successfully.
```

## no aeroscout-detection

To disable detection of AeroScout RFID Tags by APs that are managed by ZoneDirector, use the following command:

```
no aeroscout-detection
```

### **Syntax Description**

---

no aeroscout-detection	Disable detection of AeroScout RFID Tags by APs
------------------------	---

---

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# no aeroscout-detection
```

The command was executed successfully.

## ekahau

To enable and set Ekahau Blink support with ERC IP and port, use the following command:

```
ekahau <ERC IP> <ERC Port>
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# ekahau 10.10.10.1 500
```

The command was executed successfully.

```
ruckus(config-services)# show
```

Services:

```
  Automatically adjust ap radio power= Disabled
```

```
  Automatically adjust ap channel= Enabled
```

```
  Channelfly works on 2.4GHz radio:
```

```
    Status= Disabled
```



```
Channelfly works on 5GHz radio:
  Status= Disabled
Run a background scan on 2.4GHz radio:
  Status= Enabled
  Time= 2000 seconds
Run a background scan on 5GHz radio:
  Status= Enabled
  Time= 2000 seconds
AeroScout RFID tag detection= Disabled
Tunnel encryption for tunneled traffic= Disabled
Block multicast traffic from network to tunnel= Block non well-
known
Block broadcast traffic from network to tunnel except ARP and
DHCP= Disabled
Tunnel Proxy ARP of tunnel WLAN:
  status= Disabled
  ageing time= 0
Packet Inspection Filter(PIF) uplink process= Disabled
Packet Inspection Filter(PIF) rate limit:
  status= Disabled
RAPS= Enabled
EKHAU settings:
  status= Enabled
  ERC IP= 10.10.10.1
  ERC port= 500
ruckus(config-services)#
```

## **no ekahau**

To disable Ekahau Blink support, use the following command:

```
no ekahau
```

## **Defaults**

Disabled

## **Example**

```
ruckus(config-services)# no ekahau
```

The command was executed successfully.

```
ruckus(config-services)#
```

## **tun-encrypt**

To enable tunnel encryption for tunneled traffic, use the following command:

```
tun-encrypt
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# tun-encrypt
```

The command was executed successfully.

## **no tun-encrypt**

To disable tunnel encryption for tunneled traffic, use the following command:

```
no tun-encrypt
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# no tun-encrypt
```

The command was executed successfully.

## **tun-block-mcast all**

To enable multicast blocking for tunneled traffic, use the following command:

```
tun-block-mcast all
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# tun-block-mcast all
```

The command was executed successfully.

```
ruckus(config-services)#
```

### **tun-block-mcast non-well-known**

To enable multicast blocking for non-well-known tunneled traffic, use the following command:

```
tun-block-mcast non-well-known
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# tun-block-mcast non-well-known
```

The command was executed successfully.

```
ruckus(config-services)#
```

### **no tun-block-mcast**

To disable blocking multicast traffic from network to tunnel, use the following command:

```
no tun-block-mcast
```

### **tun-block-bcast**

To enable broadcast blocking for tunneled traffic, use the following command:

```
tun-block-bcast
```

### **Defaults**

Disabled

### **Example**

```
ruckus(config-services)# tun-block-bcast
```

The command was executed successfully.

```
ruckus(config-services)#
```

## no tun-block-bcast

To disable blocking broadcast traffic from network to tunnel except ARP and DHCP, use the following command:

```
no tun-block-bcast
```

## tun-proxy-arp

To enable proxy ARP service for tunneled traffic, use the following command:

```
tun-proxy-arp <NUMBER>
```

## Defaults

Disabled

## Example

```
ruckus(config-services)# tun-proxy-arp 1000
```

The command was executed successfully.

```
ruckus(config-services)#
```

## no tun-proxy-arp

To disable Proxy ARP for the tunneled WLAN, use the following command:

```
no tun-proxy-arp
```

## tun-ip-ageing

To set ageing time for IP/IPv6 table, use the following command:

```
tun-ip-ageing <NUMBER>
```

## pif

To enable Packet Inspection Filter and set rate limiting threshold, use the following command:

```
pif [uplink-proc | rate-limit <NUMBER>]
```

## Syntax Description

---

pif	Enable Packet Inspection Filter
uplink-proc	Enable uplink process of Packet Inspection Filter

---

---

rate-limit	Enable and set Broadcast Neighbor Discovery Packets (ARP and ICMPv6 Neighbor Solicit) rate limit threshold.
<NUMBER>	Rate limiting threshold for PIF feature.

---

### **Example**

```
ruckus(config-services)# pif uplink-proc
The command was executed successfully.
ruckus(config-services)# pif rate-limit 1000
The command was executed successfully.
ruckus(config-services)# show
Services:
  Automatically adjust ap radio power= Disabled
  Automatically adjust ap channel= Enabled
  Channelfly works on 2.4GHz radio:
    Status= Disabled
  Channelfly works on 5GHz radio:
    Status= Disabled
  Run a background scan on 2.4GHz radio:
    Status= Enabled
    Time= 20 seconds
  Run a background scan on 5GHz radio:
    Status= Enabled
    Time= 20 seconds
  AeroScout RFID tag detection= Disabled
  Tunnel encryption for tunneled traffic= Enabled
  Block multicast traffic from network to tunnel= Disabled
  Block broadcast traffic from network to tunnel except ARP and
  DHCP= Disabled
  Tunnel Proxy ARP of tunnel WLAN:
    status= Disabled
  Packet Inspection Filter(PIF) uplink process= Enabled
  Packet Inspection Filter(PIF) rate limit:
    status= Enabled
    rate limit= 1000
ruckus(config-services)#
```

## no pif

To disable uplink process of packet inspection filter or disables Broadcast Neighbor Discovery Packets (ARP and ICMPv6 Neighbor Solicit), use the following command:

```
no pif [uplink-proc | rate-limit]
```

### Example

```
ruckus(config-services)# no pif uplink-proc
```

The command was executed successfully.

```
ruckus(config-services)# no pif rate-limit
```

The command was executed successfully.

```
ruckus(config-services)#
```

## show

To display the current service settings, use the following command:

```
show
```

### Syntax Description

---

show	Display the current service settings
------	--------------------------------------

---

### Defaults

None.

### Example

```
ruckus(config-services)# show
```

Services:

```
  Automatically adjust ap radio power= Disabled
```

```
  Automatically adjust ap channel= Enabled
```

```
  Channelfly works on 2.4GHz radio:
```

```
    Status= Disabled
```

```
  Channelfly works on 5GHz radio:
```

```
    Status= Disabled
```

```
  Run a background scan on 2.4GHz radio:
```

```
    Status= Enabled
```

```
    Time= 2000 seconds
```

```
  Run a background scan on 5GHz radio:
```

```

    Status= Enabled
    Time= 2000 seconds
    AeroScout RFID tag detection= Disabled
    Tunnel encryption for tunneled traffic= Disabled
    Block multicast traffic from network to tunnel= Block non well-
known
    Block broadcast traffic from network to tunnel except ARP and
DHCP= Disabled
    Tunnel Proxy ARP of tunnel WLAN:
        status= Disabled
        ageing time= 0
    Packet Inspection Filter(PIF) uplink process= Disabled
    Packet Inspection Filter(PIF) rate limit:
        status= Disabled
ruckus(config-services)#

```

## Configure WIPS Commands

Use the `wips` commands to configure Wireless Intrusion Prevention settings. To run these commands, you must first enter the `config-wips` context.

### wips

Use the following command to enter the `config-wips` context and configure WIPS settings:

```
wips
```

### Syntax Description

<code>help</code>	Shows available commands
<code>history</code>	Shows a list of previously run commands
<code>end</code>	Saves changes, and the exits the <code>config-wips</code> context
<code>exit</code>	Saves changes, and the exits the <code>config-wips</code> context
<code>no &lt;WORD&gt;</code>	Disable WIPS services

protect-excessive-wireless-request	Enables protecting the wireless network against excessive wireless requests
temp-block-auth-failed-client time <NUMBER>	Temporarily block wireless clients with repeated authentication failures for the specified time (in seconds)
rogue-report <[all]   [malicious <ssid-spoofing   same-network   user-blocked   mac-spoofing]>	Enables report rogue devices in ZD event log. all: Report all rogue devices. malicious [ssid-spoofing] [same-network] [user-blocked] [mac-spoofing]: Report particular malicious type.
malicious-report	Enables protecting the network from malicious rogue access points
rogue-dhcp-detection	Enables rogue DHCP server detection
show	Displays the WIPS settings

### Example

```
ruckus(config)# wips
ruckus(config-wips)# show
Protect my wireless network against excessive wireless requests=
Disabled
Temporarily block wireless clients with repeated authentication
failures:
Status= Enabled
Time= 30 seconds
Report rogue devices in ZD event log= Enabled
Protect the network from malicious rogue access points= Disabled
Rogue DHCP server detection= Enabled
ruckus(config-wips)# temp-block-auth-failed-client time 30
The command was executed successfully.
ruckus(config-wips)# rogue-report all
The command was executed successfully.
ruckus(config-wips)# rogue-report malicious same-network
The command was executed successfully.
ruckus(config-wips)# rogue-dhcp-detection
The command was executed successfully.
ruckus(config-wips)# no rogue-dhcp-detection
The command was executed successfully.
```



```
ruckus(config-wips)# no rogue-report
The command was executed successfully.
ruckus(config-wips)# show
  Protect my wireless network against excessive wireless requests=
  Disabled
  Temporarily block wireless clients with repeated authentication
  failures:
    Status= Enabled
    Time= 30 seconds
  Report rogue devices in ZD event log= Disabled
  Protect the network from malicious rogue access points= Disabled
  Rogue DHCP server detection= Disabled
ruckus(config-wips)#
```

## Configure Email Server Commands

Use the `email-server` commands to configure email server settings. To run these commands, you must first enter the `config-email-server` context.

### email-server

Use the following command to enter the `config-email-server` context and configure email server settings:

```
email-server
```

### *Syntax Description*

help	Shows available commands.
history	Shows a list of previously run commands.
abort	Exits the <code>config-sms-server</code> context without saving changes.
end	Saves changes, and the exits the <code>config-sms-server</code> context.
exit	Saves changes, and the exits the <code>config-sms-server</code> context.
quit	Exits the <code>config-sms-server</code> context without saving changes.

enable	Enables the E-Mail server.
from <WORD>	Sets the E-Mail from for email server.
smtp-server-name <WORD>	Sets the smtp server name for email server.
smtp-server-port <NUMBER>	Sets the smtp server port for email server.
smtp-auth-name <WORD>	Sets the smtp authentication user name for email server.
smtp-auth-password <WORD>	Sets the smtp authentication password for email server.
smtp-wait-time	Sets the smtp server wait time (in seconds).
tls-smtp-encryption tls	Enables TLS of smtp encryption for email server.
tls-smtp-encryption starttls	Enables starttls in the TLS of smtp encryption for email server.
no enable	Disables the email server setting.
no tls-smtp-encryption tls	Disables TLS of smtp encryption for email server.
no tls-smtp-encryption starttls	Disables starttls in the TLS of smtp encryption for email server.
show	Shows email server settings.

### Example

```
ruckus(config)# email-server
ruckus(config-email-server)# enable
ruckus(config-email-server)# from example@example.com
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-email-server)# smtp-server-name smtp.example.com
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-email-server)# smtp-server-port 587
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-email-server)# smtp-auth-name johndoe
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-email-server)# smtp-auth-password password
The command was executed successfully. To save the changes, type
'end' or 'exit'.
```

```
ruckus(config-email-server)# tls-smtp-encryption tls
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-email-server)# tls-smtp-encryption starttls
The command was executed successfully. To save the changes, type
'end' or 'exit'.
ruckus(config-email-server)# show
Email Server:
  Status= Enabled
  E-mail From = example@example.com
  SMTP Server Name= smtp.example.com
  SMTP Server Port= 587
  SMTP Authentication Username= johndoe
  SMTP Authentication Password= *****
  SMTP Encryption Options:
    TLS= Enabled
    STARTTLS= Enabled

ruckus(config-email-server)# end
The Email server settings have been updated.
Your changes have been saved.
ruckus(config)#
```

## Configure SMS Server Commands

Use the `sms-server` commands to configure SMS server settings. To run these commands, you must first enter the `config-sms-server` context.

### **sms-server**

Use the following command to enter the `config-sms-server` context and configure SMS server settings:

```
sms-server
```

### ***Syntax Description***

---

help	Shows available commands.
history	Shows a list of previously run commands.

---

abort	Exits the config-sms-server context without saving changes.
end	Saves changes, and the exits the config-sms-server context.
exit	Saves changes, and the exits the config-sms-server context.
quit	Exits the config-sms-server context without saving changes.
twilio	Configures SMS server settings for twilio. Enters ruckus(config-sms-server-twilio)#
clickatell	Configures SMS server settings for clickatell. Enters ruckus(config-sms-server-clickatell)#
account-sid <WORD>	Sets the account sid for twilio of sms server
auth-token <WORD>	Sets the auth token for twilio of sms server
from-phonenumber <WORD>	Sets the from phonenumber for twilio of sms server
user-name <WORD>	Sets the user name for clickatell of sms server
password <WORD>	Sets the password for clickatell of sms server
api-id <WORD>	Sets the api id for clickatell of sms server
show	Displays the SMS server settings.

### **Example**

```
ruckus(config)# sms-server
```

```
ruckus(config-sms-server)# twilio
```

```
ruckus(config-sms-server-twilio)# account-sid example1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-sms-server-twilio)# auth-token token1
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-sms-server-twilio)# from-phonenumber  
111222333444555
```

The command was executed successfully. To save the changes, type 'end' or 'exit'.

```
ruckus(config-sms-server-twilio)# end
```

The SMS server settings have been updated.

```
Your changes have been saved.  
ruckus(config-sms-server)# show  
SMS Server:  
  Server Type= twilio  
  Account SID= example1  
  Auth Token= token1  
  From PhoneNumber= 111222333444555
```

```
ruckus(config-sms-server)# end  
The SMS server settings have been updated.  
Your changes have been saved.  
ruckus(config)#
```

## **no sms-server**

To disable SMS server settings, use the following command:

```
no sms-server
```

### ***Example***

```
ruckus(config)# no sms-server  
The SMS server settings have been updated.  
ruckus(config)#
```

## **Configure mDNS (Bonjour) Commands**

Use the following commands to configure mDNS (Bonjour Gateway) service.

### **mdnsproxy**

Use the following command to enable mDNS proxy (Bonjour Gateway) service:

```
mdnsproxy [zd|ap]
```

### **no mdnsproxy**

Use the following command to disable mDNS proxy (Bonjour Gateway) service:

```
no mdnsproxy [zd|ap]
```

## mdnsproxyrule

Use the following command to create a new Bonjour Gateway rule or modify an existing rule, and enter the `config-mdnsproxyrule` context:

```
mdnsproxyrule <ID>
```

## no mdnsproxyrule

Use the following command to delete a Bonjour Gateway rule:

```
no mdnsproxyrule <ID>
```

## Configuring a Bonjour Policy

The following commands can be used from within the `config-bonjourpolicy` context to configure the Bonjour policy.

### bonjour-policy

To create or edit a Bonjour policy, use the following command:

```
bonjour-policy <WORD>
```

### Syntax Description

help	Shows available commands
history	Shows a list of previously run commands
no mdnsproxyrule	Delete mDNSproxy rule
mdnsproxyrule <ID>	Add/update mDNSproxy rules
note <NOTE>	Rule comments
end	Save the current rule and quit
exit	Save the current rule and quit
abort	Discard the current rule and quit
quit	Discard the current rule and quit

### Example

```
ruckus(config)# bonjour-policy bonjour1  
ruckus(config-bonjourpolicy)# note bonjourpolicy1  
ruckus(config-bonjourpolicy)# end
```

```
Your changes have been saved.  
ruckus(config)# show bonjour-policy  
bonjour-policy:  
  ID: 1  
  Name: bonjour1  
  Description: bonjourpolicy1  
  rule:  
ruckus(config)#
```

## no bonjour-policy

To delete a Bonjour policy, use the following command:

```
no bonjour-policy <WORD>
```

## Configuring mDNS Proxy Rules

The following commands can be used from within the `config-mdnsproxyrule` context to configure the Bonjour Gateway bridge service rule.

### Syntax Description

help	Shows available commands
history	Shows a list of previously run commands
service <Service-Name>	Service name in ? list, or new bonjour rule
from-vlan <VLAN-From>	VLAN from
to-vlan <VLAN-to>	VLAN to
note <NOTE>	Rule comments
show	Show the current edited rule
end	Save the current rule and quit
abort	Discard the current rule and quit
quit	Discard the current rule and quit

### Example

```
ruckus(config-bonjourpolicy)# mdnsproxyrule 1
```

```
ruckus(config-policyrule)# service AirDisk
ruckus(config-policyrule)# from-vlan 220
ruckus(config-policyrule)# to-vlan 1
ruckus(config-policyrule)# note "share printer to vlan1"
ruckus(config-policyrule)# end
ruckus(config-bonjourpolicy)# end
ruckus(config)# show bonjour-policy
bonjour-policy:
  ID: 1
  Name: bonjour1
  Description: bonjourpolicy1
  rule:
    1:
      mdnsservice: AirDisk
      from_vlan: br0.220
      to_vlan: br0
      Notes: share printer to vlan1
ruckus(config)#
```



# Using Debug Commands

## 4

In this chapter:

- [Debug Commands Overview](#)
- [General Debug Commands](#)
- [Show Commands](#)
- [Accessing a Remote AP CLI](#)
- [Working with Debug Logs and Log Settings](#)
- [Remote Troubleshooting](#)
- [AP Core Dump Collection](#)
- [Script Execution](#)

## Debug Commands Overview

This section describes the commands that you can use to debug ZoneDirector and connected APs, and to configure debug log settings. From the privileged commands context, type **debug** to enter the debug context. To show a list of commands available from within the debug context, type `help` or `?`.

## General Debug Commands

The following section describes general debug commands can be executed from within the debug context.

### help

Shows available commands.

### list-all

List all available commands.

### history

Shows a list of previously run commands.

### quit

Exits the debug context.

### fw\_upgrade

To upgrade the controller's firmware, use the following command:

```
fw_upgrade <protocol>://<server ip|server name>/<path/
image name> [-f]
fw_upgrade OPTIONS
```

### *Syntax Description*

<code>fw_upgrade</code>	Upgrade the controller's firmware
<code>&lt;protocol&gt;</code>	Protocol for image transfer (FTP, TFTP, HTTP, KERMIT)

<code>&lt;OPTIONS&gt;</code>	<p>-p: protocol</p> <p>-s: server IP address or name</p> <p>-n: image name with path on the server</p> <p>-f: non-verbose mode</p> <p>-h: fw_upgrade help message</p>
------------------------------	---

### **Defaults**

None.

### **Example**

```
ruckus# debug
ruckus(debug)# fw_upgrade ftp://<user>:<password>@<server ip>/
<image file>
```

### **delete-station**

To deauthorize the station with the specified MAC address, use the following command.

```
delete-station <MAC>
```

### **Syntax Description**

<code>delete-station</code>	Delete the station with the specified MAC address
<code>&lt;MAC&gt;</code>	The MAC address of the station that will be deleted

### **Defaults**

None.

### **Example**

```
ruckus# debug
ruckus(debug)# delete-station 00:10:77:01:00:01
The command was executed successfully.
```

## restart-ap

To restart the device with the specified MAC address, use the `restart ap` command.

```
restart-ap <MAC>
```

### Syntax Description

<code>restart-ap</code>	Restart the device with the specified MAC address
<code>&lt;MAC&gt;</code>	The MAC address of the device to be restarted

### Defaults

None.

### Example

```
ruckus# debug
ruckus(debug)# restart-ap 00:13:92:EA:43:01
The command was executed successfully.
```

## wlaninfo

Configures and enables debugging of WLAN service settings. Enter `wlaninfo` without arguments to see all options.

```
wlaninfo <OPTIONS>
```

### Syntax Description

<code>wlaninfo</code>	Enable logging of WLAN info
<code>&lt;OPTIONS&gt;</code>	Configure WLAN debug information options

### Defaults

None.

### Example

```
ruckus(debug)# wlaninfo -W -x
```

```
WLAN svc "Rhastah1" (id=1):
  WLAN ID = 0, ref_cnt = 7
  SSID = "Rhastah1" enabled
  Apply to 11a and 11g/b radios
  Closed system = No, Privacy = Enabled, ACL enabled Guest-WLAN = No
  WISPr-WLAN = No
  Access Policy = 0/0, Web Auth = No, grace period = 0 (0 means
  disable), max clients = 100
  WMM = enabled priority = 0 uplink = DISABLE downlink = DISABLE
  Cipher = Clear Text Local bridging = Enabled, DHCP relay = Disabled,
  vlan = 1, dvlan = Disabled, bgscan = Enabled
  Proxy ARP = Disabled (IE:Disabled)
  wep key index = 0, wep key len = 0
  PAP message authenticator = Enabled, EAP-Failure = Disabled
  Device Policy = 0, Precedence = 1
  Smart Roam = Disabled Roam-factor = 1
  Hotspot2.0--WLAN = No (id=0)
  Num of VAP deployed: 6
    VAP: 04:4f:aa:0c:b1:0c, number of stations = 0
    VAP: 04:4f:aa:0c:b1:08, number of stations = 0
    VAP: c0:c5:20:3b:91:fc, number of stations = 1
    VAP: c0:c5:20:3b:91:f8, number of stations = 0
    VAP: c4:10:8a:1f:d1:fc, number of stations = 1
    VAP: c4:10:8a:1f:d1:f8, number of stations = 0
  ACL 1 (System): default=Allowed system-wide=yes
  Auth Policy:
    Auth Algorithms:RSN/PSK  RSN/Dynamic PSK
    Auth Server Type: None
    WPA Verson: WPA2
    WPA Auth and Key Managment: WPA PSK
    WPA PSK Pass Phrase:password
    WPA PSK Prev Pass Phrase:
    WPA PSK Pass Phrase (Hex):
      31306173 68613130
    WPA PSK:
      6aa94bac df5346ac ecc7d38f a14a6dbf
      7ba6f6f8 df2a4943 b23c9655 ac4f33de
    WPA Prev PSK:
      00000000 00000000 00000000 00000000
```

```

00000000 00000000 00000000 00000000
GTK life time = 28800 seconds, GTK Life size = 2000 Kpkts
GMK life time = 86400 seconds, Strict Rekey = No
WPA Group Cipher Suites:0x00000010
    CCMP
WPA Pairwise Cipher Suites:0x00000010
    CCMP
NASID Type: = wlan-bssid
PMK Cache Time: = 43200
PMK Cache for Reconnect: = enabled
Roaming Acct-Inerim-Update: = disabled
Called-Station-Id-type: 0
Classification: enabled
UDP Heuristic Classification: enabled
Directed Multicast: enabled
IGMP Snooping: enabled
MLD Snooping: disabled
ToS Classification: enabled
Dot1p Classification: disabled
Multicast Filter: disabled
Directed Threshold: 5
Priority: Voice:0   Video:2   Data:4   Background:6
Force DHCP: disabled   Timeout:10

*** Total WLAN Entries: 1 ***
ruckus(debug)#

```

## **save\_debug\_info**

Saves debug information.

```
save_debug_info <IP-ADDR> <FILE-NAME>
```

### ***Syntax Description***

save_debug_info	Save debug log file
<IP-ADDR>	The destination IP address
<FILE-NAME>	The destination file name

## Defaults

None.

## Example

```
ruckus(debug)# save_debug_info 192.168.11.26 log.log  
Creating debug info file ...  
Done  
Sending debug info file to "log.log@192.168.11.26" ...  
...  
ruckus(debug)#
```

## save-config

Upload the configuration file to the designated TFTP site.

```
save-config <IP-ADDR> <FILE-NAME>
```

## Syntax Description

save-config	Upload the configuration file
<IP-ADDR>	The destination IP address
<FILE-NAME>	The destination file name

## Defaults

None.

## Example

```
ruckus(debug)# save-config 192.168.11.26 config.log  
Creating backup config file  
Done  
Uploading backup config file  
...  
ruckus(debug)#
```

## emfd-malloc-stats

Show uclibc malloc statistics.

### Example

```
ruckus(debug)# emfd-malloc-stats
===== [pid=350] Sat Feb 15 15:58:42 2014
total bytes allocated           = 2691072
total bytes in use              = 2471920
total bytes freed               = 219152
total allocated mmap space     = 311296
number of free chunks          = 18
number of fastbin blocks       = 0
space in freed fastbin blocks  = 0
bin[ 1]: chunk_num= 1, list_len= 1, alloc_bytes= 4152,
min_chunk[1]= 4152, max_chunk[1]= 4152
bin[ 3]: chunk_num= 3, list_len= 3, alloc_bytes= 72,
min_chunk[1]= 24, max_chunk[1]= 24
bin[ 4]: chunk_num= 1, list_len= 1, alloc_bytes= 32,
min_chunk[1]= 32, max_chunk[1]= 32
bin[ 5]: chunk_num= 4, list_len= 4, alloc_bytes= 160,
min_chunk[1]= 40, max_chunk[1]= 40
bin[ 6]: chunk_num= 1, list_len= 1, alloc_bytes= 48,
min_chunk[1]= 48, max_chunk[1]= 48
bin[10]: chunk_num= 1, list_len= 1, alloc_bytes= 80,
min_chunk[1]= 80, max_chunk[1]= 80
bin[14]: chunk_num= 1, list_len= 1, alloc_bytes= 112,
min_chunk[1]= 112, max_chunk[1]= 112
bin[45]: chunk_num= 1, list_len= 1, alloc_bytes= 2928,
min_chunk[1]= 2928, max_chunk[1]= 2928
bin[49]: chunk_num= 1, list_len= 1, alloc_bytes= 5168,
min_chunk[1]= 5168, max_chunk[1]= 5168
bin[51]: chunk_num= 2, list_len= 2, alloc_bytes= 14952,
min_chunk[1]= 7248, max_chunk[2]= 7704
bin[52]: chunk_num= 1, list_len= 1, alloc_bytes= 8208,
min_chunk[1]= 8208, max_chunk[1]= 8208
ruckus(debug)#
```



# Show Commands

This section describes the show commands available within the debug context.

## show ap

Displays a list of all approved devices.

```
show ap
```

### Syntax Description

---

show ap	Display a list of all approved APs
---------	------------------------------------

---

### Defaults

None.

### Example

```
ruckus(debug) # show ap
AP:
  ID:
    1:
      MAC Address= 04:4f:aa:0d:b1:00
      Model= zf7962
      Approved= Yes
      Device Name= 7962-MAP
      ...
      ...
ruckus(debug) #
```

## show station

Displays a list of all connected stations (or clients).

```
show station
```

### Syntax Description

---

show station	Show all connected stations
--------------	-----------------------------

---

## Defaults

None.

## Example

```
ruckus(debug)# show station
Clients List:
Client:
  MAC Address= 6c:62:6d:1b:e3:00
  User Name=
  IP Address= 192.168.11.11
  IPv6 Address=
  Access Point= 04:4f:aa:0c:b1:00
  WLAN= Ruckus1
  Channel= 1
  Signal (dB)= 53

Client:
  MAC Address= 00:22:fb:ad:1b:2e
  User Name=
  IP Address= 192.168.11.7
  IPv6 Address=
  Access Point= 04:4f:aa:0c:b1:00
  WLAN= Ruckus1
  Channel= 165
  Signal (dB)= 42

ruckus(debug)#
```

## show logs

Displays a list of debug log components.

```
show logs
```

## Syntax Description

---

<code>show logs</code>	Display debug log components
------------------------	------------------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(debug)# show logs
Debug Logs:
  All= Enabled
  Sys-mgmt= Enabled
  Mesh= Enabled
  Web-auth= Enabled
  Rf-mgmt= Enabled
  Radius= Enabled
  Hotspot-srv= Enabled
  Aps= Enabled
  Net-mgmt= Enabled
  802.1x= Enabled
  Web-svr= Enabled
  802.11= Enabled
  Dvlan= Enabled
  Smart-redundancy= Enabled
  Debug logs of specified MAC address:
    Status= Disabled
ruckus(debug)#
```

### **show remote-troubleshooting**

Shows remote-troubleshooting status.

```
show remote-troubleshooting
```

### ***Syntax Description***

---

```
show remote-      Display remote troubleshooting status
troubleshooting
```

---

### **Defaults**

None.

### **Example**

```
ruckus(debug)# show remote-troubleshooting
```

Ruckus CA troubleshooting is stopped!

The server addr is: None

```
ruckus(debug)#
```

### **ps**

Displays information about all processes that are running (ps -aux).

```
ps
```

### **Syntax Description**

---

```
ps                  Display a list of all running processes
```

---

### **Defaults**

None.

### **Example**

```
ruckus(debug) # ps
```

PID	PPID	USER	VSZ	STAT	COMMAND
1	0	ruckus	1200	S	init
2	1	ruckus	0	SWN	[ksoftirqd/0]
3	1	ruckus	0	SW	[watchdog/0]
4	1	ruckus	0	SW<	[events/0]
5	1	ruckus	0	SW<	[khelper]

```

    6      1 ruckus      0 SW<  [kthread]
    7      6 ruckus      0 SW<  [kblockd/0]
    8      6 ruckus      0 SW<  [khubd]
    9      6 ruckus      0 SW   [pdflush]
   10     6 ruckus      0 SW   [pdflush]
   12     6 ruckus      0 SW<  [aio/0]
   11     1 ruckus      0 SW   [kswapd0]
   13     1 ruckus      0 SW   [mtdblockd]
   14     6 ruckus      0 SW<  [scsi_eh_0]
   15     6 ruckus      0 SW<  [usb-storage]
   17     6 ruckus      0 SW<  [V54_bodygard/0]
   18     1 ruckus      0 SW   [pktgen/0]
   29     6 ruckus      0 SW<  [reiserfs/0]
  104     1 ruckus     956 S   /usr/sbin/in.tftpd -l -s /etc/
airespider-images
  110     1 ruckus     660 S   /bin/wd_feeder
  242     1 ruckus    2572 S   /bin/emf_repo_flashsync monitor 15
  243     1 ruckus     944 S   ttylogd
  246     1 ruckus      0 SW<  [uif-246]
  260     1 ruckus    14492 S   stamgr -d3 -t0
  266    260 ruckus    14492 S   stamgr -d3 -t0
  267    266 ruckus    14492 S <  stamgr -d3 -t0
  268    266 ruckus    14492 S   stamgr -d3 -t0
  269     1 ruckus     2268 S   apmgr
  277    269 ruckus     2268 S   apmgr
  278    277 ruckus     2268 S <  apmgr
  299     1 ruckus    19564 S   emfd
  316    299 ruckus    19564 S   emfd
  317    316 ruckus    19564 S   emfd
  318    316 ruckus    19564 S   emfd
  322     1 ruckus    1108 S   /usr/sbin/dropbear -e /bin/login.sh
-r /etc/air
  328     1 ruckus     1188 S   /bin/sh /bin/login.sh
  329     1 ruckus     1188 S   /bin/sh /bin/tacmon.sh
  331     1 ruckus      676 S   /bin/rhttpd
  332     1 ruckus     1140 S <  /bin/zapd
  333     1 ruckus     1100 S <  /bin/clusterD
  334    328 ruckus      856 S   /bin/login
  335    329 ruckus      680 S   /bin/tacmon -i 30 -r 15

```

```

347      1 ruckus      808 S    /bin/tsyslogd -r -h -n --rotate=7
368    277 ruckus     2268 S <  apmgr
369    277 ruckus     2268 S <  apmgr
572      1 ruckus     1184 S    /sbin/udhcpd -i br0 --pidfile=/var/
run/udhcpd.p
580    316 ruckus     19564 S    emfd
612    316 ruckus     19564 S    emfd
616    316 ruckus     19564 S    emfd
622    316 ruckus     19564 S    emfd
624    299 ruckus     6132 S <  webs &
625    316 ruckus     19564 S    emfd
637    624 ruckus     6132 S    webs &
638    637 ruckus     6132 S <  webs &
639    637 ruckus     6132 S <  webs &
640    637 ruckus     6132 S <  webs &
641    637 ruckus     6132 S <  webs &
642    637 ruckus     6132 S    webs &
655    637 ruckus     6132 S <  webs &
656    637 ruckus     6132 S <  webs &
20503  316 ruckus     19564 S    emfd
30679      1 ruckus     2672 S    /usr/sbin/vsftpd /etc/vsftpd2.conf
10220  322 ruckus     1184 S    /usr/sbin/dropbear -e /bin/login.sh
-r /etc/air
10221 10220 ruckus     1188 S    /bin/sh /bin/login.sh
10222 10221 ruckus     856 S    /bin/login
10223 10222 ruckus     7972 S    ruckus_cli2
10426 10223 ruckus     1188 S    sh -c /bin/ps -aux
10427 10426 ruckus     1188 R    /bin/ps -aux
ruckus (debug) #

```

## Accessing a Remote AP CLI

The following command is used to access the command line interface of a connected AP and execute AP CLI commands from ZoneDirector. Configuration changes made through the AP CLI may be overwritten by ZoneDirector settings if the AP is restarted or reconnects to ZoneDirector.

## remote\_ap\_cli

Use the `remote_ap_cli` command to access an AP remotely and execute AP CLI commands.

```
remote_ap_cli [-q] {-a ap_mac | -A } "cmd arg1 arg2 .."
```

### Syntax Description

<code>remote_ap_cli</code>	Execute CLI commands in a remote AP
<code>-q</code>	Do not display results
<code>-a</code>	Specify AP by MAC address
<code>ap_mac</code>	The AP's MAC address
<code>-A</code>	All connected APs
<code>cmd</code>	AP CLI command
<code>arg</code>	AP CLI command argument

### Example

```
ruckus(debug)# remote_ap_cli -A "get director"
---- Command 'rkscli -c "get director "' executed at
c0:c5:20:3b:91:f0
----- ZoneDirector Info -----
Primary Controller   : n/a
Secondary Controller : n/a
DHCP Opt43 Code     : 3

The information of the most recent Zone Director:
[1] 192.168.40.100

AP is under management of ZoneDirector: 192.168.40.100 /
c0:c5:20:18:97:c1,
Currently AP is in state: RUN
OK
---- Command 'rkscli -c "get director "' executed at
c4:10:8a:1f:d1:f0
----- ZoneDirector Info -----
Primary Controller   : n/a
```

```
Secondary Controller : n/a  
DHCP Opt43 Code      : 3
```

The information of the most recent Zone Director:

```
[1] 192.168.40.100
```

```
AP is under management of ZoneDirector: 192.168.40.100 /  
c0:c5:20:18:97:c1,
```

```
Currently AP is in state: RUN
```

```
OK
```

```
---- Command Execution Summary:
```

```
    success: 2
```

```
    failure: 0
```

```
    total: 2
```

```
ruckus(debug)#
```



# Working with Debug Logs and Log Settings

This section describes the commands that you can use to configure and review ZoneDirector debug logs.

## logs all

Enables debug logs of all debug components.

---

**NOTE** Running this command can place considerable load on the system. If your ZoneDirector is already under load, running this command could potentially cause errors resulting in a reboot. In general, only use this command when working with Ruckus support to troubleshoot an issue.

---

## Syntax Description

---

logs all	Enable logging of all debug components
----------	--

---

## Example

```
ruckus(debug)# logs all
The command was executed successfully.
ruckus(debug)# show logs
Debug Logs:
  All= Enabled
  Sys-mgmt= Enabled
  Mesh= Enabled
  Web-auth= Enabled
  Rf-mgmt= Enabled
  Radius= Enabled
  Hotspot-srv= Enabled
  Aps= Enabled
  Net-mgmt= Enabled
  802.1x= Enabled
  Web-svr= Enabled
  802.11= Enabled
  Dvlan= Enabled
  Smart-redundancy= Enabled
  Client-association= Enabled
```

```

    Debug logs of specified MAC address:
    Status= Disabled
ruckus(debug)#

```

## no logs all

Disables debug logs of all debug components.

### *Syntax Description*

no logs	Disable debug logs
all	Disable all log components

### *Example*

```

ruckus(debug)# no logs all
The command was executed successfully.
ruckus(debug)#

```

## logs comp sys-mgmt

Enables debug logs of system management components.

### *Syntax Description*

logs	Enable debug logs
comp sys-mgmt	Component system management

### *Example*

```

ruckus(debug)# logs comp sys-mgmt
The command was executed successfully.
ruckus(debug)# show logs
Debug Logs:
    All= Disabled
    Sys-mgmt= Enabled

```

```
Mesh= Disabled
Web-auth= Disabled
Rf-mgmt= Disabled
Radius= Disabled
Hotspot-srv= Disabled
Aps= Disabled
Net-mgmt= Disabled
802.1x= Disabled
Web-svr= Disabled
802.11= Disabled
Dvlan= Disabled
Smart-redundancy= Disabled
Client-association= Disabled
Debug logs of specified MAC address:
  Status= Disabled
ruckus(debug)#
```

### **no logs comp sys-mgmt**

Disables debug logs of system management components.

### **logs comp mesh**

Enables debug logs of mesh components.

### **no logs comp mesh**

Disables debug logs of mesh components.

### **logs comp web-auth**

Enables debug logs of web authentication components.

### **no logs comp web-auth**

Disables debug logs of web authentication components.

### **logs comp rf-mgmt**

Enables debug logs of RF management components.

**no logs comp rf-mgmt**

Disables debug logs of RF management components.

**logs comp radius**

Enables debug logs of radius components.

**no logs comp radius**

Disables debug logs of radius components.

**logs comp hotspot-srv**

Enables debug logs of hotspot services components.

**no logs comp hotspot-srv**

Disables debug logs of hotspot services components.

**logs comp aps**

Enables debug logs of AP components.

**no logs comp aps**

Disables debug logs of access points components.

**logs comp net-mgmt**

Enables debug logs of network management components.

**no logs comp net-mgmt**

Disables debug logs of network management components.

**logs comp 802.1x**

Enables debug logs of 802.1x components.

**no logs comp 802.1x**

Disables debug logs of 802.1x components.

**logs comp web-svr**

Enables debug logs of web server components.

### **no logs comp web-svr**

Disables debug logs of web server components.

### **logs comp 802.11**

Enables debug logs of 802.11 components.

### **no logs comp 802.11**

Disables debug logs of 802.11 components.

### **logs comp dvlan**

Enables debug logs of dynamic VLAN components.

### **no logs comp dvlan**

Disables debug logs of dynamic vlan components.

### **logs comp smart-redundancy**

Enable Smart Redundancy component debug logs.

### **no logs comp smart-redundancy**

Disable Smart Redundancy component debug logs.

### **logs comp bonjour-gateway**

Enable Bonjour Gateway debug logs.

### **no logs comp bonjour-gateway**

Disable Bonjour Gateway debug logs.

### **logs comp mdnsd**

Enable bonjour mdnsd debug logs.

### **no logs comp mdnsd**

Disable bonjour mdnsd debug logs.

### **logs comp client-association**

Enable client association debug logs.

## no logs comp client-association

Disable client association debug logs.

## logs mac

Enables and sets filter running logs based on specified mac address.

```
logs mac <MAC>
```

### *Syntax Description*

logs	Enable debug logs
mac	Filter logs by specific MAC address
<MAC>	The MAC address of the device to be filtered

### *Example*

```
ruckus(debug)# logs mac 04:4f:aa:0c:b1:00  
The command was executed successfully.  
ruckus(debug)#
```

## no logs mac

Disables MAC address filtering on running logs.

### *Syntax Description*

no logs	Disable debug logs
mac	Filter by MAC address

### *Example*

```
ruckus(debug)# no logs mac  
The command was executed successfully.  
ruckus(debug)#
```

## logs play

Starts displaying logs on console.

---

**CAUTION!** Running this command can place considerable load on the system. If your ZoneDirector is already under load, running this command could potentially cause errors resulting in a reboot. In general, only use this command when working with Ruckus support to troubleshoot an issue.

---

### Syntax Description

logs	Enable debug logs
play	Start log play

### Example

```
ruckus(debug)# logs play
ruckus(debug)# [Feb 15 05:53:30][EMFD][debug]jobService-
Func():Executing job[user auth attempt_hash_autoexpire] at
1329285210...
[Feb 15 05:53:30][EMFD][debug]jobServiceFunc():Executing job at
1329285210...Done
[Feb 15 05:53:30][EMFD][debug]jobServiceFunc():Executing
job[station auth attempt_hash_autoexpire] at 1329285210...
[Feb 15 05:53:30][EMFD][debug]jobServiceFunc():Executing job at
1329285210...Done
[Feb 15 05:53:33][STAMgr][debug]acsrvc_thread():ACSRVC rcv AP
04:4f:aa:0c:b1:00, IP= 192.168.11.6, IPv6=fc00::1
...
...
ruckus(debug)# no logs play
ruckus(debug)#
```

## no logs play

Stops displaying logs on console.

## Syntax Description

---

no logs	Disable debug logs
play	Stop log play

---

### Example

```
ruckus(debug)# logs play
ruckus(debug)# [Feb 15 05:53:30][EMFD][debug]jobService-
Func():Executing job[user auth attempt_hash_autoexpire] at
1329285210...
[Feb 15 05:53:30][EMFD][debug]jobServiceFunc():Executing job at
1329285210...Done
[Feb 15 05:53:30][EMFD][debug]jobServiceFunc():Executing
job[station auth attempt_hash_autoexpire] at 1329285210...
[Feb 15 05:53:30][EMFD][debug]jobServiceFunc():Executing job at
1329285210...Done
[Feb 15 05:53:33][STAMgr][debug]acsrvc_thread():ACSRVC rcv AP
04:4f:aa:0c:b1:00, IP= 192.168.11.6, IPv6=fc00::1
...
...
ruckus(debug)# no logs play
ruckus(debug)#
```

## Remote Troubleshooting

This section describes remote troubleshooting commands.

### remote-troubleshooting server

To set the remote troubleshooting server IP address, use the following command:

```
remote-troubleshooting server <IP-ADDR>
```

### remote-troubleshooting start

Enables remote troubleshooting.

## Syntax Description



---

<code>remote- troubleshooting</code>	Remote troubleshooting
<code>start</code>	Start remote troubleshooting

---

### ***Defaults***

None.

### ***Example***

```
ruckus (debug) # remote-troubleshooting start
```

```
ruckus (debug) #
```

### **remote-troubleshooting stop**

Disables remote troubleshooting.

### ***Syntax Description***

---

<code>remote- troubleshooting</code>	Remote troubleshooting
<code>stop</code>	Stop remote troubleshooting

---

### ***Defaults***

None.

### ***Example***

```
ruckus (debug) # remote-troubleshooting stop
```

```
ruckus (debug) #
```

## radius-stats-wlan

Show web-auth WLAN radius statistics bins.

## radius-stats-authsvr

Show web-auth WLAN radius statistics bins.

# AP Core Dump Collection

This section lists the AP core dump commands.

## collect\_ap\_coredump

Enable AP core dump collection.

```
collect_ap_coredump [all|<MAC>]
```

### *Syntax Description*

collect_ap_core dupm	Collect AP core dump
all	Collect core dump from all connected APs
<MAC>	Specific AP MAC address

### *Defaults*

None.

### *Example*

```
ruckus(debug)# collect_ap_coredump all
---- Command 'apmgrinfo --coredump y ' executed at 04:4f:aa:0c:b1:00
start reporting coredump to ZD!
---- Command 'apmgrinfo --coredump y ' executed at 00:24:82:3f:14:60
start reporting coredump to ZD!
---- Command Execution Summary:
      success: 2
      failure: 0
```

```

        total: 2
rm: cannot remove '/etc/airespider-images/firmwares/ap-dump/*': No
such file or directory
sh: codump_server: not found
start collecting AP's coredump !
ok
ruckus(debug)#

```

## no collect\_ap\_coredump

Disable AP core dump collection.

### Syntax Description

---

no	Stop collecting AP core dump
collect_ap_core	
dump	

---

### Defaults

None.

### Example

```

ruckus(debug)# no collect_ap_coredump all
---- Command 'apmgrinfo --coredump n ' executed at 04:4f:aa:0c:b1:00
stop reporting coredump to ZD!
---- Command 'apmgrinfo --coredump n ' executed at 00:24:82:3f:14:60
stop reporting coredump to ZD!
---- Command Execution Summary:
        success: 2
        failure: 0
        total: 2
rm: cannot remove '/etc/airespider-images/firmwares/ap-dump/*': No
such file or directory
stop collecting AP's coredump !
ok
ruckus(debug)#

```

## Script Execution

This section lists the commands that can be executed from the script context. The script context must be entered from the debug context.

### **script**

Enters the script context from the debug context. You must first enter the script context before executing a script.

```
script
```

### ***Syntax Description***

---

<code>script</code>	Enter the script context
---------------------	--------------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(debug)# script  
ruckus(script)#
```

### **quit**

Exit the script context.

```
quit
```

### ***Syntax Description***

---

<code>quit</code>	Exit the script context
-------------------	-------------------------

---

### ***Defaults***

None.

### **Example**

```
ruckus(script)# quit
```

```
ruckus(debug)#
```

### **list**

List all available scripts.

```
list
```

### **Syntax Description**

---

<code>list</code>	List all available scripts
-------------------	----------------------------

---

### **Defaults**

None.

### **Example**

```
ruckus(script)# list -a
```

```
Index           Scripts
```

```
1               .version.sh
```

```
ruckus(script)#
```

### **del**

Deletes a script.

### **info**

Display script help file

```
info
```

## ***Syntax Description***

---

<code>info</code>	Display script information
-------------------	----------------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(script)# info
info <file>
ruckus(script)#
```

### ***exec***

Execute script.

```
exec <file> {parameter}
```

## ***Syntax Description***

---

<code>exec</code>	Excecute the script
-------------------	---------------------

---

### ***Defaults***

None.

### ***Example***

```
ruckus(script)# exec
exec <file> {parameter}
ruckus(script)#
```

# Index

## Numerics

11n-only 167  
80211w-pmf 356  
802dot11d 343

## A

aaa 103  
aaa all 22  
aaa name 24  
abort 98, 159, 161, 179, 185, 204,  
211, 276, 289, 360, 368, 382, 387,  
408, 446, 460, 472  
access-ctrl 376  
accs-net-type chargeable-public 441  
accs-net-type free-public 441  
accs-net-type personal-device 441  
accs-net-type private 441  
accs-net-type private-with-guest 441  
accs-net-type test-or-experimental 441  
accs-net-type wildcard 441  
acct-server 331, 419  
acct-server interim-update 332, 420  
acl 202  
acl dvcpcy 353  
acl end 212  
acl prece 353  
acl quit 213  
acl role-based-access-ctrl 353  
active-wired-client 88  
act-threshold 244  
add-mac 208  
ad-global-catalog 104  
adj-threshold 242  
admin 109, 375  
admin-dn 105  
admin-password 105  
admission-control 120, 168, 171, 337  
adv-gas cb-delay 445  
adv-gas dos-detect 445  
adv-gas dos-maxreq 445  
adv-gas rsp-buf-time 445  
adv-gas rsp-limit 445  
aeroscout-detection 479  
alarm 84, 460  
alarm-event 468  
allow-indoor 248  
anqp-3gpp-info 447  
ap all 26  
ap devname 29  
AP\_group model-specific port settings  
179  
ap mac 31, 35  
ap-auto-approve 153  
ap-group 160  
ap-group all 33  
ap-group name 35  
ap-management-vlan 151  
app-denial-policy 231  
application 237  
application-visibility 343  
apply-policy-group 343  
ap-policy 36, 150  
app-port-mapping 237  
asra 441  
asra dns 441  
asra enrollment 441  
asra http-https 441  
asra http-https url 441  
asra terms 441  
authentication guest-pass 391  
auth-method chap 104  
auth-method pap 104  
auth-server 110  
auth-server local 416  
auth-server name 416  
auth-server name mac-bypass 417  
auth-server name mac-bypass mac-addr-  
format 418  
auth-server name no-mac-bypass 417  
auth-server with-fallback 111  
auto-adjust-ap-channel 475  
auto-adjust-ap-power 474  
auto-channel-selection 167  
autonomous 298  
auto-proxy 344  
auto-recovery 156

## B

- background-scan 478
- backup 105
- backup-ip-addr 105
- backup-port 105
- backup-radius-secret 105
- band-balancing 240, 327
- beacon-interval 296, 457
- bgscan 324
- Bonjour 493
- bonjour 279
- bonjour-gateway 114
- bonjour-policy 494
- bss-minrate 337
- bypasscna 273

## C

- called-station-id-type 295
- cband-channels 127, 177
- channel 120, 167
- channelfly 476
- channelflyoff 164
- channelization 120, 167
- channel-mode 248
- channel-optimization 248
- channel-range 120
- clickatell 492
- client fingerprinting 342
- client-isolation 325, 421
- collect\_ap\_coredump 522
- config 18
- config wlan dot1x authentication encryption wpa2 algorithm TKIP auth-server 318
- conn-cap esp 444
- conn-cap ftp 442
- conn-cap http 443
- conn-cap icmp 442
- conn-cap ikev2 443
- conn-cap ipsec-vpn 443
- conn-cap pptp-vpn 443
- conn-cap ssh 443
- conn-cap tls-vpn 443
- conn-cap voip-tcp 443
- conn-cap voip-udp 443
- consecutive-drop-packet 105
- contact 262
- country code 247
- creating a WLAN 364

- current-active-clients 75
- custm-conn-cap 444

## D

- debug 18
- del 525
- delete station 499
- del-mac 209
- description 108, 115, 162, 207, 215, 218, 225, 230, 239, 295, 363, 371, 397, 403, 429, 435, 447
- destination 224
- destination address 220, 224, 399, 405, 431
- destination port 220, 224, 399, 405, 431
- destination-IP 236
- destination-port 237
- device fingerprinting 342
- devinfo 230
- devname 114
- dhcp 106
- dhcp all 25
- dhcp name 25
- dhcp-relay 339
- disable 18
- disable wifi0 245
- disable wifi1 245
- disable-dgaf 355
- disabling NTP client 254
- disabling SNMP agent 286, 287
- disabling SNMP traps 287, 288
- displaying interface settings 252
- domain-name 104, 447
- dot11-country-code 247
- dot1x 192
- dot1x acctsvr 147, 181, 194
- dot1x authentication encryption wep-64 auth-server 321
- dot1x authentication encryption wpa algorithm AES auth-server 315
- dot1x authentication encryption wpa algorithm TKIP auth-server 315
- dot1x authentication encryption wpa2 algorithm AES auth-server 317
- dot1x authsvr 147, 181, 193
- dot1x eap-type EAP-SIM auth-server 313
- dot1x eap-type PEAP auth-server 314
- dot1x mac-auth-bypass 148, 181, 194



- dot1x none 322
- dot1x supplicant mac 149, 181, 196
- dot1x supplicant password 149, 195
- dot1x supplicant user-name 181
- dot1x supplicant username 148, 195
- dot1x supplicant user-name password 181
- dot1x wep-128 auth-server 322
- dot1x wpa algorithm auto auth-server 316
- dot1x wpa2 algorithm auto auth-server 318
- dot1x wpa-mixed algorithm AES auth-server 319
- dot1x wpa-mixed algorithm TKIP auth-server 320
- dot1x-mac none 323
- dvccpy 227
- dvlan 146
- dynamic-certs 81
- dynamic-psk enable 347
- dynamic-psk passphrase-len 348
- dynamic-psk type 348
- dynamic-psk-expiration 293, 349
- dynamic-psks 80
- dynamic-vlan 334

## E

- eap-method 449
- eap-method auth-info 450
- eap-method eap-mthd 449
- ekahau 480
- e-mail 463
- email-server 489
- emfd-malloc-stats 504
- enable wifi0 245
- enable wifi1 246
- encoding 449
- encryption-TLS 105
- end 98, 179, 186, 204, 212, 218, 275, 289, 360, 369, 388, 408, 446, 461, 473
- ethinfo 43
- event 468
- event-log-level 272
- events-activities 83
- exec 526
- exit 18, 98, 159, 161, 180, 186, 205, 212, 218, 275, 289, 361, 369, 388, 409, 446, 461, 473

- extant-gain 121
- external-antenna 126, 177, 178

## F

- facility 270
- fan-out-threshold 457
- first 108
- flexmaster 259
- force-dhcp 339
- force-dhcp-timeout 340
- from 464
- from-vlan 495
- ftp 255
- ftp-anon 254
- ft-roaming 324
- full-name 385
- fw\_upgrade 498

## G

- gateway 249, 258, 259
- gps 116
- grace-period 330, 415
- group 117
- group-attributes 371
- grp-search 104
- guest-access 298, 387
- guestpass-duration 388
- guest-passes 81
- guest-pass-generation 374
- guestpass-notification 389
- guestpass-reauth 388
- guestpass-share-number 389
- guestpass-sponsor 389
- guestpass-sponsor-auth-server 389
- guestpass-sponsor-number 389
- guestpass-terms-and-conditions 390
- guest-vlan 146

## H

- headroom 245
- help 18, 98, 179, 446, 498
- hessid 435
- hessid-use-bssid 435
- heuristics classification video packet-  
octet-count 278
- heuristics classification voice packet-  
octet-count 278
- heuristics no-classification video packet-

- octet-count 278
- heuristics no-classification voice packet-octet-count 278
- heuristics video inter-packet-gap 277
- heuristics video packet-length 277
- heuristics voice inter-packet-gap 278
- heuristics voice packet-length 278
- hide ssid 336
- history 18, 98, 179, 446, 498
- hops-warn-threshold 456
- hostname 248
- hotspot 298, 407
- hotspot all 60
- hotspot name 61
- hotspot\_redirect\_https 201
- hs20 298
- hs20op 433
- hs20sp 445
- hs-caps operating-class-indication 2.4 445
- hs-caps operating-class-indication 5 445
- hs-caps operating-class-indication dual-band 445
- https-redirect 325

## I

- icmpv6-type 406, 427
- icmpv6-type Any 224
- icmpv6-type number 225
- ignor-unauth-stats 356
- import-aplist 159
- inactivity-timeout 333
- info 525
- interface 249
- internal-heater 127, 177
- internet-option 435
- intrusion-prevention 432
- ip 118, 239
- ip addr 250, 257
- IP address 250
- IP address mode 251
- ip enable 249
- ip mode 251
- ip mode DHCP 118
- ip mode keep 118
- ip mode static 118
- ip name-server 250
- ip route gateway 249
- ip-addr 104

- ip-addr-type ipv4 double-nated 441
- ip-addr-type ipv4 not-avail 441
- ip-addr-type ipv4 port-double 442
- ip-addr-type ipv4 port-restricted 441
- ip-addr-type ipv4 port-single 441
- ip-addr-type ipv4 public 441
- ip-addr-type ipv4 single-nated 441
- ip-addr-type ipv4 unknown 442
- ip-addr-type ipv6 avail 442
- ip-addr-type ipv6 not-avail 442
- ip-addr-type ipv6 unknown 442
- ipmode 128, 164
- ipv6 119
- ipv6 addr 253, 259
- ipv6 enable 253
- ipv6 mode 253
- ipv6 mode auto 119
- ipv6 mode keep 119
- ipv6 mode manual 119
- ipv6 name-server 253
- ipv6 route gateway 253

## K

- key-attribute 105

## L

- l2acl all 53
- l2acl name 54
- l3acl 210
- l3acl all 57
- l3acl name 58
- l3acl-ipv6 210, 222
- l3acl-ipv6 all 57
- l3acl-ipv6 name 58
- lan 140, 180, 188
- lan guest-vlan 180
- lan dot1x 146
- lan dot1x auth-mac-based 180
- lan dot1x auth-port-based 180
- lan dot1x disabled 180
- lan dot1x supplicant 180
- lan dvlan 197
- lan dvlan disabled 146, 180
- lan dvlan enabled 146, 180
- lan guest-vlan 197
- lan member 143, 180, 191
- lan opt82 144, 192
- lan opt82 disabled 180

- lan opt82 enabled 180
- lan qos 197
- lan qos igmp-snooping 181, 197
- lan qos mld-snooping 181, 197
- lan tunnel 144
- lan tunnel disabled 180
- lan tunnel enabled 180
- lan untag 142, 180, 190
- lan uplink 141, 180, 189
- license 85
- limit 349
- limit-dpsk 349
- limited mode 16
- limited-zd-discovery 154
- limited-zd-discovery keep-ap-setting 156
- limited-zd-discovery prefer-primary-zd 155
- list 525
- list-all 498
- lldp 130, 199
- load-balancing 241, 326
- location 116, 263
- location-id 421
- location-name 422
- location-services 102, 162
- login-page 413
- login-warning 283
- logo 18
- logs all 513
- logs comp 802.11 517
- logs comp 802.1x 516
- logs comp aps 516
- logs comp bonjour-gateway 517
- logs comp client-association 517
- logs comp dvlan 517
- logs comp hotspot-srv 516
- logs comp mDNSd 517
- logs comp mesh 515
- logs comp net-mgmt 516
- logs comp radius 516
- logs comp rf-mgmt 515
- logs comp smart-redundancy 517
- logs comp sys-mgmt 514
- logs comp web-auth 515
- logs comp web-svr 516
- logs mac 518
- logs play 519

## M

- mac 239
- mac authentication encryption none auth-server 307
- mac authentication encryption wep-128 key key-id auth-server 313
- mac authentication encryption wep-64 key key-id auth-server 312
- mac authentication encryption wpa passphrase algorithm AES auth-server 308
- mac authentication encryption wpa passphrase algorithm TKIP auth-server 308
- mac authentication encryption wpa2 passphrase algorithm AES auth-server 309
- mac authentication encryption wpa2 passphrase algorithm TKIP auth-server 310
- mac wpa-mixed passphrase algorithm AES auth-server 310
- mac wpa-mixed passphrase algorithm TKIP auth-server 311
- mac-addr-format 352
- malicious-report 488
- max clients 342
- max-clients 177, 342
- mcast-filter 335
- mdnsproxy 493
- mdnsproxy from-vlan 495
- mdnsproxy service 495
- mdnsproxy to-vlan 495
- mdnsproxyrule 494
- member 143, 182
- member add mac 182
- member mac move-to name 184
- member mac move-to system-default 184
- mesh 453
- mesh info 78
- mesh mode 122
- mesh mode auto 123
- mesh mode disable 123
- mesh mode mesh-ap 123
- mesh mode root-ap 123
- mesh topology 79
- mesh uplink-selection 123
- mesh uplink-selection add-mac 123
- mesh uplink-selection auto 123
- mesh uplink-selection del-mac 123
- mesh uplink-selection manual 123

- mesh-uplink-selection dynamic 459
- mesh-uplink-selection static 459
- mgmt-acl 274
- mgmt-acl all 46
- mgmt-acl name 46
- mgmt-acl-ipv6 275
- mgmt-acl-ipv6 all 46
- mgmt-acl-ipv6 name 46
- mgmt-if 257
- mgmt-if-ipv6 258
- mgmt-tx-rate 297, 458
- mode allow 208, 215, 223
- mode deny 209, 216, 223
- model 176
- model c-band channels 177
- model external-antenna 177, 178
- model internal-heater 177
- model max-clients 177
- model poe-out 177
- model port-setting 176, 179, 184
- model radio-band 177
- model spectra-analysis 177
- model status-leds 176
- model usb-software 177
- model-specific port settings 179
- monitor 19
- monitor ap mac 91
- monitor current-active-clients 93
- monitor current-active-clients-mcs-info 94
- monitor sysinfo 94
- move-ap 158

## N

- nai-realm 447
- name 108, 109, 206, 214, 225, 239, 276, 289, 298, 362, 370, 388, 411, 435, 447, 448
- name password 110
- nasid-type 328
- netmask 236
- new-trigger 244
- no 80211w-pmf 356
- no 802dot11d 343
- no access-ctrl 377
- no acct-server 333, 419
- no acl 203
- no ad-global-catalog 104
- no admin 375

- no admission-control 337
- no adv-gas dos-detect 435
- no aeroscout-detection 480
- no alarm 460
- no anqp-3gpp-info 447
- no ap 113
- no ap-auto-approve 153
- no ap-group 160
- no ap-management-vlan 152
- no app-denial-policy 233
- no application-visibility 343
- no asra 435
- no asra dns 435
- no asra enrollment 435
- no asra http-https 435
- no asra http-https-url 435
- no asra terms 435
- no authentication 390
- no auth-server 111
- no auto-adjust-ap-channel 476
- no auto-adjust-ap-power 475
- no auto-proxy 345
- no auto-recovery 157
- no background-scan 478
- no backup 104
- no band-balancing 327
- no bgscan 324
- no blocked-client 202
- no bonjour 280
- no bonjour-gateway 115
- no bonjour-policy 495
- no bss-minrate 338
- no bypasscna 273
- no cband-channels-override 128
- no channelfly 477
- no channelflyoff 166
- no channelflyoff-override 166
- no collect\_ap\_coredump 523
- no custom-conn-cap 435
- no description 115, 162
- no detect-fanout 457
- no detect-hops 456
- no devname 114
- no dhcp 108
- no dhcp-relay 339
- no disable-dgaf 355
- no domain-name 446
- no dot1x 196, 198
- no dot1x acctsvr 181, 198
- no dot1x authsvr 181, 198

no dot1x mac-auth-bypass 181, 198  
no dvcpcy 231, 351  
no dynamic-psk 348  
no dynamic-vlan 335  
no ekahau 481  
no encryption-TLS 104  
no event 470  
no external-antenna-override 126  
no flexmaster 260  
no force-dhcp 340  
no friendly-name 435  
no ftp 255  
no ftp-anon 255  
no ft-roaming 324  
no gateway 258, 259  
no gps 116  
no grace-period 331, 415  
no grp-search 104  
no guest-access 387  
no guest-pass-generation 374  
no guestpass-reauth 389  
no guestpass-sponsor 389  
no guestpass-terms-and-conditions 390  
no hessid 434  
no hide ssid 336  
no hotspot 407  
no hotspot\_redirect\_https 201  
no hs20op 434  
no hs20sp 446  
no hs-caps operating-class-indication 435  
no https-redirection 325  
no ignor-unauth-stats 356  
no internal-heater-override 127  
no internet-option 434  
no intrusion-prevention 433  
no ip 253  
no ipmode-override 129, 164  
no ipv6 120, 253  
no l2acl 350  
no l3acl 211, 351  
no l3acl-ipv6 351  
no lan 141, 181, 188  
no lan qos 198  
no lan qos igmp-snooping 181, 198  
no lan qos mld-snooping 181, 198  
no limit-dpsk 349  
no limited-zd-discovery 155  
no limited-zd-discovery keep-ap-setting 156  
no limited-zd-discovery prefer-primary-zd 156  
no lldp-override 131  
no load-balancing 241, 326  
no location 117  
no location-services 103, 163  
no login-warning 284  
no logs all 514  
no logs comp 802.11 517  
no logs comp 802.1x 516  
no logs comp aps 516  
no logs comp bonjour-gateway 517  
no logs comp client-association 518  
no logs comp dvlan 517  
no logs comp hotspot-srv 516  
no logs comp mDNSd 517  
no logs comp mesh 515  
no logs comp net-mgmt 516  
no logs comp radius 516  
no logs comp rf-mgmt 516  
no logs comp smart-redundancy 517  
no logs comp sys-mgmt 515  
no logs comp web-auth 515  
no logs comp web-svr 517  
no logs mac 518  
no logs play 519  
no mac-addr-format 352  
no mcast-filter 335  
no mdnspoxy 493  
no mdnspoxyrule 494  
no mgmt-acl 274  
no mgmt-acl-ipv6 275  
no mgmt-if 257, 258  
no model-setting 176  
no nai-realm 446  
no northbound 261  
no ntp 254  
no ofdm-only 337  
no onboarding 390  
no option82 341  
no pap-authenticator 328  
no pif 486  
no pmk-cache 345  
no pmk-cache-for-reconnect 345  
no poe-out-override 125  
no port-setting 179  
no prece 227  
no proxy-arp 356  
no qos 276  
no qos classification 354

no qos directed-multicast 354  
no qos heuristics-udp 354  
no qos igmp-query v2 175  
no qos igmp-query v3 176  
no qos igmp-snooping 354  
no qos mld-query v1 175  
no qos mld-query v2 175  
no qos mld-snooping 355  
no qos tos-classification 355  
no radio 121  
no radio 2.4 11n-only-override 173  
no radio 2.4 admission-control 173  
no radio 2.4 admission-control-override 173  
no radio 2.4 channelization-override 173  
no radio 2.4 channel-override 173  
no radio 2.4 channel-range-override 173  
no radio 2.4 spectralink-compatibility-override 173  
no radio 2.4 tx-power-override 173  
no radio 2.4 wlan-group-override 173  
no radio 5 11n-only-override 174  
no radio 5 admission-control 174  
no radio 5 admission-control-override 174  
no radio 5 channelization-override 174  
no radio 5 indoor channel-override 174  
no radio 5 indoor channel-range-override 174  
no radio 5 outdoor channel-override 174  
no radio 5 outdoor channel-range-override 174  
no radio 5 spectralink-compatibility-override 174  
no radio 5 tx-power-override 174  
no radio 5 wlan-group-override 174  
no radio-band-override 129  
no radius-encryption 104  
no raps 476  
no rate-limit 230, 352, 378  
no restrict-access-order 395, 425  
no restrict-access-order-ipv6 401, 426  
no roam-consortium 446  
no roaming-acct-interim-update 346  
no role 368  
no role-based-access-ctrl 351  
no rrm-neigh-report 325  
no rule 233, 236, 237, 239  
no rule-order 216, 223  
no second 108  
no self-service 388  
no send eap-failure 327  
no service-provider 434  
no session-limit-unauth-stats 286  
no session-stats-resv 285  
no session-timeout 414  
no smartclient 412  
no smart-redundancy 256  
no smart-roam 339  
no sms-server 493  
no snmp-agent 286, 287  
no snmp-trap 287  
no snmp-trap-ap 269  
no snmpv2 286  
no snmpv2-ap 264  
no snmpv2-trap 287  
no snmpv3 287  
no snmpv3-trap 288  
no specify-os-type-access 378  
no specify-wlan-access 373  
no sta-info-extraction 342  
no static-route 282  
no static-route-ipv6 283  
no status-leds-override 125  
no stp 247  
no syslog 269  
no syslog-ap 274  
no telnetd 280  
no term-of-use 391  
no timeout 158  
no tls-smtp-encryption 467  
no tun-block-bcast 484  
no tun-block-mcast 483  
no tun-encrypt 482  
no tunnel mode 338  
no tun-proxy-arp 484  
no upnp 292  
no usb-port-override 125  
no usb-software 128  
no usb-software-override 126  
no user 381  
no venue-group-type 434  
no venue-name 130  
no vlan-pool 381  
no vlanpool 351  
no vlan-qos 157  
no walled-garden 423  
no wan-metrics at-cap 435  
no wan-metrics sym 435  
no web authentication 330

- no whitelist 238, 326
- no wlan-group 359
- no zero-it-activation 346
- northbound 260
- not-allow-indoor 248

## O

- ofdm-only 337
- onboarding 390
- open authentication encryption wep-128
  - key key-id 306
- open authentication encryption wep-64
  - key key-id 306
- open authentication encryption wpa passphrase algorithm AES 300
- open authentication encryption wpa passphrase algorithm auto 302
- open authentication encryption wpa passphrase algorithm TKIP 301, 302
- open authentication encryption wpa2 passphrase algorithm AES 303
- open authentication encryption wpa2 passphrase algorithm TKIP 303
- open none 300
- open wpa passphrase algorithm auto 302
- open wpa2 passphrase algorithm auto 304
- open wpa-mixed passphrase algorithm auto 305
- opt82 144
- option82 340
- order 218, 224, 397, 403, 429
- os-type-allowed all 377
- os-type-allowed specify 377

## P

- pap-authenticator 328
- passphrase 455
- password 385
- peer-addr 255
- pif 484
- ping 18
- pmk-cache 345
- pmk-cache-for-reconnect 345
- poe-out 125, 177
- port 104
- port settings 179
- port-setting 134, 176, 179

- prece 225
- priority 271
- privileged mode 16
- protect-excessive-wireless-request 488
- protocol 221, 224, 237, 400, 406, 432
- proxy-arp 355
- ps 508

## Q

- qos 175, 277
- qos classification 354
- qos directed-multicast 354
- qos directed-threshold 355
- qos heuristics-udp 354
- qos igmp-query 175
- qos igmp-query v2 175
- qos igmp-query v3 175
- qos igmp-snooping 354
- qos mld-query 175
- qos mld-query v1 175
- qos mld-query v2 175
- qos mld-snooping 354
- qos priority high 355
- qos priority low 355
- qos tos-classification 355
- quit 18, 98, 159, 161, 180, 187, 199, 205, 213, 275, 289, 362, 370, 388, 410, 446, 462, 474, 498, 524

## R

- radio 120, 167
- radio 2.4 120
- radio 2.4 11n-only Auto 170
- radio 2.4 11n-only N-only 170
- radio 2.4 admission-control 171
- radio 2.4 auto-channel-selection four-channel 170
- radio 2.4 auto-channel-selection three-channel 170
- radio 2.4 channel auto 169
- radio 2.4 channel number 169
- radio 2.4 channelization auto 169
- radio 2.4 channelization number 170
- radio 2.4 channel-range 171
- radio 2.4 spectralink-compatibility 171
- radio 2.4 tx-power 1/2 170
- radio 2.4 tx-power 1/4 170
- radio 2.4 tx-power 1/8 170

- radio 2.4 tx-power Auto 170
- radio 2.4 tx-power Full 170
- radio 2.4 tx-power Min 170
- radio 2.4 tx-power Num 170
- radio 2.4 wlan-group 171
- radio 5 120
- radio 5 11n-only Auto 172
- radio 5 11n-only N-only 172
- radio 5 admission-control 173
- radio 5 channel auto 171
- radio 5 channel number 172
- radio 5 channelization auto 172
- radio 5 channelization number 172
- radio 5 indoor channel auto 171
- radio 5 indoor channel number 171
- radio 5 indoor channel-range 171
- radio 5 outdoor channel auto 171
- radio 5 outdoor channel number 171
- radio 5 outdoor channel-range 171
- radio 5 spectralink-compatibility 173
- radio 5 tx-power 1/2 172
- radio 5 tx-power 1/4 172
- radio 5 tx-power 1/8 172
- radio 5 tx-power Auto 172
- radio 5 tx-power Full 172
- radio 5 tx-power Min 172
- radio 5 tx-power Num 172
- radio 5 wlan-group 173
- radio-band 129, 177
- radius-encryption 104
- radius-encryption tls 104
- radius-secret 105
- radius-stats-authsvr 522
- radius-stats-wlan 522
- raps 476
- rate-limit 230, 351
- rate-limit uplink 378
- rate-limit uplink downlink 378
- read-only community 263
- read-write community 263
- reboot 18
- reconnect-primary-interval 105
- redirect 392
- re-generate-private-key 200
- remote\_ap\_cli 511
- remote-troubleshooting server 520
- remote-troubleshooting start 520
- remote-troubleshooting stop 521
- request-timeout 105
- reset 18

- reset radius-statistics 89
- restart-ap 500
- restore 200
- restrict-access-order 396, 424
- restrict-access-order-ipv6 401, 425
- restrict-type 276
- restrict-type range ip-range 290
- restrict-type single ip-addr 289
- restrict-type subnet ip-subnet 290
- retry-count 105
- roam-consortium 447
- roaming-acct-interim-update 345
- ro-community 263
- rogue-devices 82
- rogue-dhcp-detection 488
- rogue-report 488
- role 367, 386
- role all 71
- role name 72
- rrm-neigh-report 325
- rule 226, 229, 233, 236, 237, 239
- rule-order 217, 223
- rw-community 263

## S

- save-config 503
- save\_debug\_info 502
- script 524
- search-filter 105
- second 108
- secret 255
- self-service 388
- send eap-failure 327
- service-provider 435
- session-limit-unauth-stats 286
- session-stats-resv 285
- session-timeout 19, 87, 414
- set-factory 18
- show 18, 108, 131, 138, 151, 159, 161, 180, 187, 206, 213, 221, 246, 252, 264, 276, 279, 284, 291, 356, 367, 378, 386, 394, 397, 402, 410, 428, 445, 447, 454, 462, 486
- show aaa 99
- show active-wired-client all 88
- show active-wired-client mac 88
- show admin 99
- show ap 99, 505
- show ap-group 101



show app-denial-policy 86, 100  
 show ap-policy 101  
 show app-port-mapping 87, 100  
 show bonjour-policy 101  
 show current-active-clients mac 76  
 show dhcp 25, 99  
 show dhcp all 25  
 show dhcp name 25  
 show dvccpy 100  
 show guest-access-service 69, 101  
 show hotspot 101  
 show hs20op 70  
 show hs20op all 62  
 show hs20op name 65  
 show hs20sp 71  
 show hs20sp all 67  
 show hs20sp name 68  
 show l2acl 99  
 show l3acl 99  
 show l3acl-ipv6 100  
 show load-balance 90  
 show load-balancing 100  
 show location-services 21, 101  
 show location-services name 22  
 show logs 506  
 show mdnsproxy 101  
 show mdnsproxyrule 101  
 show mgmt-acl 99  
 show mgmt-acl-ipv6 99  
 show performance 39  
 show performance ap-radio2-4 39  
 show performance ap-radio5 40  
 show performance station 41  
 show prece 100  
 show radius-statistics 89  
 show remote-troubleshooting 507  
 show role 100  
 show static-route 99  
 show static-route-ipv6 99  
 show station 505  
 show support-entitle 284  
 show usb-software 101  
 show user 101  
 show user all 74  
 show user name 74  
 show user-defined-app 86, 100  
 show vlan-pool 73, 100  
 show whitelist 100  
 show whitelist all 55, 56  
 show wlan 48, 100  
 show wlan-group 100  
 shutdown 18  
 smartclient 412  
 smartclient info 412  
 smartclient secure http 412  
 smartclient secure https 412  
 smartclient wispr-only secure http 412  
 smartclient wispr-only secure https 412  
 smart-redundancy 255  
 smart-roam 339  
 sms-server 491  
 smtp-auth-name 465  
 smtp-auth-password 466  
 smtp-server-name 464  
 smtp-server-port 465  
 smtp-wait-time 466  
 SNMP RO 263  
 SNMP RW 263  
 snmp-trap 288  
 snmp-trap-format 267  
 snmpv2 261  
 snmpv2-ap 264  
 snmpv2-trap 268  
 snmpv3 265  
 snmpv3-trap 268  
 specify-os-type-access 377  
 specify-wlan-access 373  
 spectra-analysis 177  
 spectra-analysis 2.4GHz 126  
 spectra-analysis 5GHz 127  
 spectralink-compatibility 168, 171, 173  
 ssid 296, 454  
 sta-info-extraction 342  
 standard-usage 298  
 start-page 413  
 static-route 281  
 static-route all 47  
 static-route name 47  
 static-route-ipv6 282  
 static-route-ipv6 all 47  
 static-route-ipv6 name 48  
 status-leds 124, 176  
 stp 247  
 strong-bypass 243  
 support-entitle 285  
 sysinfo 37  
 syslog 270  
 syslog notifications 269  
 sysstats 42  
 system 247

## T

- tacplus-secret 105
- tacplus-service 104
- techsupport 44
- telnetd 280
- temp-block-auth-failed-client 488
- term-of-use 392
- timeout 158
- tls-smtp-encryption 467
- tos classification background 279
- tos classification data 279
- tos classification video 278
- tos classification voice 279
- to-vlan 495
- trap server 288
- tun-block-bcast 483
- tun-block-mcast all 482
- tun-block-mcast non-well-known 483
- tun-encrypt 482
- tun-ip-ageing 484
- tunnel 144
- tunnel mode 338
- tunnel-mtu 279
- tun-proxy-arp 484
- twilio 492
- tx-power 120, 167
- type 104, 230, 298
- type ad 104
- type allow 219, 224, 398, 404, 430
- type autonomous 299
- type deny 219, 224, 398, 404, 430
- type guest-access 299
- type hotspot 299
- type hs20 299
- type ldap 104
- type radius-acct 104
- type radius-auth 104
- type standard-usage 299
- type tacplus-auth 104

## U

- uplink 141
- upnp 291
- usb-port 125, 176
- usb-software 85, 128, 177
- user 381
- user-defined-app 235
- user-name 384

## V

- venue-group-type assembly 435
- venue-group-type unspecified 435
- venue-name 130
- vlan 230, 253, 258, 259, 334, 378
- vlan-pool 379
- vlanpool 352
- vlan-qos 157

## W

- walled-garden 422
- wan-metrics at-cap 442
- wan-metrics downlink-load 442
- wan-metrics downlink-speed 442
- wan-metrics link-stat down 442
- wan-metrics link-stat test 442
- wan-metrics link-stat up 442
- wan-metrics lmd 442
- wan-metrics sym 442
- wan-metrics uplink-load 442
- wan-metrics uplink-speed 442
- weak-bypass 243
- web authentication 329
- web-auth 329
- welcome-text 393
- whitelist 238, 326, 421
- wips 487
- wlan 294, 364
- WLAN description 295
- WLAN SSID 296
- wlan vlan override none 366
- wlan vlan override tag 366
- wlan-allowed 372
- wlan-group 120, 167, 171, 173, 359
- wlan-group all 51
- wlan-group name 52
- wlaninfo 500
- wlan-service 120

## Z

- zero-it 292
- zero-it-activation 346
- zero-it-auth-server 292
- ZoneDirector
  - gateway 249
  - IP address 250
  - IP address mode 251
  - name server 250



Copyright © 2006-2015. Ruckus Wireless, Inc.  
350 West Java Dr. Sunnyvale, CA 94089. USA  
[www.ruckuswireless.com](http://www.ruckuswireless.com)