


**Part No. 060304-10, Rev. B
August 2010**

OmniSwitch 6250 Network Configuration Guide

Alcatel·Lucent 

www.alcatel-lucent.com

**This user guide documents release 6.6.2 of the OmniSwitch 6250 Series.
The functionality described in this guide is subject to change without notice.**

Copyright © 2010 by Alcatel-Lucent. All rights reserved. This document may not be reproduced in whole or in part without the express written permission of Alcatel-Lucent.

Alcatel-Lucent® and the Alcatel-Lucent logo are registered trademarks of Alcatel-Lucent. Xylan®, OmniSwitch®, OmniStack®, and Alcatel-Lucent OmniVista® are registered trademarks of Alcatel-Lucent.

OmniAccess™, Omni Switch/Router™, PolicyView™, RouterView™, SwitchManager™, VoiceView™, WebView™, X-Cell™, X-Vision™, and the Xylan logo are trademarks of Alcatel-Lucent.

This OmniSwitch product contains components which may be covered by one or more of the following U.S. Patents:

- U.S. Patent No. 6,339,830
- U.S. Patent No. 6,070,243
- U.S. Patent No. 6,061,368
- U.S. Patent No. 5,394,402
- U.S. Patent No. 6,047,024
- U.S. Patent No. 6,314,106
- U.S. Patent No. 6,542,507
- U.S. Patent No. 6,874,090



**26801 West Agoura Road
Calabasas, CA 91301
(818) 880-3500 FAX (818) 880-3505
support@ind.alcatel.com**

**US Customer Support—(800) 995-2696
International Customer Support—(818) 878-4507
Internet—service.esd.alcatel-lucent.com**

Contents

	About This Guide	xxxv
	Supported Platforms	xxxv
	Who Should Read this Manual?	xxxvi
	When Should I Read this Manual?	xxxvi
	What is in this Manual?	xxxvi
	What is Not in this Manual?	xxxvii
	How is the Information Organized?	xxxvii
	Documentation Roadmap	xxxviii
	Related Documentation	xxxix
	User Manual CD	xl
	Technical Support	xl
Chapter 1	Configuring Ethernet Ports	1-1
	In This Chapter	1-1
	Ethernet Specifications	1-2
	Ethernet Port Defaults (All Port Types)	1-2
	Non-Combo Port Defaults	1-3
	Combo Ethernet Port Defaults	1-3
	Ethernet Ports Overview	1-4
	OmniSwitch Series Combo Ports	1-4
	Valid Port Settings on OmniSwitch 6250 Series Switches	1-5
	10/100/1000 Crossover Supported	1-5
	Autonegotiation Guidelines	1-5
	Flow Control and Autonegotiation	1-6
	Setting Ethernet Parameters for All Port Types	1-7
	Setting Trap Port Link Messages	1-7
	Enabling Trap Port Link Messages	1-7
	Disabling Trap Port Link Messages	1-7
	Resetting Statistics Counters	1-8
	Enabling and Disabling Interfaces	1-8
	Configuring Flood Rate Limiting	1-9
	Flood Only Rate Limiting	1-9
	Multicast Flood Rate Limiting	1-9
	Configuring the Peak Flood Rate Value	1-10
	Configuring a Port Alias	1-11
	Configuring Maximum Frame Sizes	1-11

Setting Ethernet Parameters for Non-Combo Ports	1-12
Setting Interface Line Speed	1-12
Configuring Duplex Mode	1-12
Configuring Inter-frame Gap Values	1-13
Configuring Autonegotiation and Crossover Settings	1-14
Enabling and Disabling Autonegotiation	1-14
Configuring Crossover Settings	1-15
Configuring Flow Control on Non-Combo Ports	1-15
Setting Ethernet Combo Port Parameters	1-17
Setting Interface Line Speed for Combo Ports	1-17
Configuring Duplex Mode for Combo Ports	1-18
Configuring Autonegotiation and Crossover for Combo Ports	1-18
Enabling and Disabling Autonegotiation for Combo Ports	1-18
Configuring Crossover Settings for Combo Ports	1-19
Configuring Flow Control on Combo Ports	1-20
Verifying Ethernet Port Configuration	1-22
Chapter 2 Managing Source Learning	2-1
In This Chapter	2-1
Source Learning Specifications	2-2
Source Learning Defaults	2-2
Sample MAC Address Table Configuration	2-3
MAC Address Table Overview	2-5
Using Static MAC Addresses	2-5
Configuring Static MAC Addresses	2-6
Static MAC Addresses on Link Aggregate Ports	2-6
Using Static Multicast MAC Addresses	2-7
Configuring Static Multicast MAC Addresses	2-7
Static Multicast MAC Addresses on Link Aggregate Ports	2-8
ASCII-File-Only Syntax	2-8
Configuring MAC Address Table Aging Time	2-9
Configuring the Source Learning Status	2-10
Displaying Source Learning Information	2-11
Chapter 3 Configuring Learned Port Security	3-1
In This Chapter	3-1
Learned Port Security Specifications	3-2
Learned Port Security Defaults	3-2
Sample Learned Port Security Configuration	3-3
Learned Port Security Overview	3-4
How LPS Authorizes Source MAC Addresses	3-5
Dynamic Configuration of Authorized MAC Addresses	3-5
Static Configuration of Authorized MAC Addresses	3-6

	Understanding the LPS Table	3-6
	Configuring Learned Port Security	3-7
	Enabling/Disabling Learned Port Security	3-7
	Configuring a Source Learning Time Limit	3-8
	Configuring the Number of Bridged MAC Addresses Allowed	3-9
	Configuring the Trap Threshold for Bridged MAC Addresses	3-9
	Configuring the Number of Filtered MAC Addresses Allowed	3-10
	Configuring Authorized MAC Addresses	3-10
	Configuring an Authorized MAC Address Range	3-10
	Selecting the Security Violation Mode	3-11
	Displaying Learned Port Security Information	3-12
Chapter 4	Configuring VLANs	4-1
	In This Chapter	4-1
	VLAN Specifications	4-2
	VLAN Defaults	4-2
	Sample VLAN Configuration	4-3
	VLAN Management Overview	4-4
	Creating/Modifying VLANs	4-5
	Adding/Removing a VLAN	4-5
	Enabling/Disabling the VLAN Administrative Status	4-6
	Modifying the VLAN Description	4-6
	Defining VLAN Port Assignments	4-7
	Changing the Default VLAN Assignment for a Port	4-7
	Configuring Dynamic VLAN Port Assignment	4-8
	Configuring VLAN Rule Classification	4-8
	Enabling/Disabling VLAN Mobile Tag Classification	4-9
	Enabling/Disabling Spanning Tree for a VLAN	4-10
	Configuring VLAN Router Interfaces	4-11
	What is Single MAC Router Mode?	4-11
	Bridging VLANs Across Multiple Switches	4-12
	Verifying the VLAN Configuration	4-13
Chapter 5	Configuring GVRP	5-1
	In This Chapter	5-1
	GVRP Specifications	5-2
	GVRP Defaults	5-2
	GARP Overview	5-3
	GVRP Overview	5-3
	Quick Steps for Configuring GVRP	5-5
	Configuring GVRP	5-7
	Enabling GVRP	5-7

	Enabling Transparent Switching	5-8
	Configuring the Maximum Number of VLANs	5-8
	Configuring GVRP Registration	5-9
	Setting GVRP Normal Registration	5-9
	Setting GVRP Fixed Registration	5-9
	Setting GVRP Forbidden Registration	5-9
	Configuring the GVRP Applicant Mode	5-10
	Modifying GVRP timers	5-10
	Restricting VLAN Registration	5-11
	Restricting Static VLAN Registration	5-12
	Restricting VLAN Advertisement	5-12
	Verifying GVRP Configuration	5-13
Chapter 6	Assigning Ports to VLANs	6-1
	In This Chapter	6-1
	Port Assignment Specifications	6-2
	Port Assignment Defaults	6-2
	Sample VLAN Port Assignment	6-3
	Statically Assigning Ports to VLANs	6-4
	Dynamically Assigning Ports to VLANs	6-4
	How Dynamic Port Assignment Works	6-5
	VLAN Mobile Tag Classification	6-5
	VLAN Rule Classification	6-8
	Configuring Dynamic VLAN Port Assignment	6-10
	Enabling/Disabling Port Mobility	6-11
	Ignoring Bridge Protocol Data Units (BPDU)	6-11
	Understanding Mobile Port Properties	6-12
	What is a Configured Default VLAN?	6-12
	What is a Secondary VLAN?	6-13
	Configuring Mobile Port Properties	6-16
	Enable/Disable Default VLAN	6-16
	Enable/Disable Default VLAN Restore	6-17
	Enable/Disable 802.1X Port-Based Access Control	6-17
	Verifying VLAN Port Associations and Mobile Port Properties	6-18
	Understanding ‘show vlan port’ Output	6-18
	Understanding ‘show vlan port mobile’ Output	6-19
Chapter 7	Configuring Port Mapping	7-1
	In This Chapter	7-1
	Port Mapping Specifications	7-2
	Port Mapping Defaults	7-2
	Quick Steps for Configuring Port Mapping	7-2
	Creating/Deleting a Port Mapping Session	7-3
	Creating a Port Mapping Session	7-3
	Deleting a User/Network Port of a Session	7-3

Deleting a Port Mapping Session	7-3
Enabling/Disabling a Port Mapping Session	7-4
Enabling a Port Mapping Session	7-4
Disabling a Port Mapping Session	7-4
Configuring a Port Mapping Direction	7-4
Configuring Unidirectional Port Mapping	7-4
Restoring Bidirectional Port Mapping	7-4
Sample Port Mapping Configuration	7-5
Example Port Mapping Overview	7-5
Example Port Mapping Configuration Steps	7-6
Verifying the Port Mapping Configuration	7-6
Chapter 8	
Defining VLAN Rules	8-1
In This Chapter	8-1
VLAN Rules Specifications	8-2
VLAN Rules Defaults	8-2
Sample VLAN Rule Configuration	8-3
VLAN Rules Overview	8-4
VLAN Rule Types	8-4
DHCP Rules	8-5
MAC Address Rules	8-5
Network Address Rules	8-5
Protocol Rules	8-5
Port Rules	8-6
Understanding VLAN Rule Precedence	8-6
Configuring VLAN Rule Definitions	8-8
Defining DHCP MAC Address Rules	8-9
Defining DHCP MAC Range Rules	8-9
Defining DHCP Port Rules	8-10
Defining DHCP Generic Rules	8-10
Defining MAC Address Rules	8-10
Defining MAC Range Rules	8-11
Defining IP Network Address Rules	8-11
Defining Protocol Rules	8-12
Defining Port Rules	8-13
Application Example: DHCP Rules	8-14
The VLANs	8-14
DHCP Servers and Clients	8-14
Verifying VLAN Rule Configuration	8-17
Chapter 9	
Configuring VLAN Stacking	9-1
In This Chapter	9-1
VLAN Stacking Specifications	9-2
VLAN Stacking Defaults	9-2

VLAN Stacking Overview	9-3
How VLAN Stacking Works	9-5
VLAN Stacking Services	9-6
Interaction With Other Features	9-7
GARP VLAN Registration Protocol (GVRP)	9-7
IP Multicast VLANs	9-7
Link Aggregation	9-8
Quality of Service (QoS)	9-8
Ring Rapid Spanning Tree Protocol (RRSTP)	9-8
Spanning Tree	9-8
Quick Steps for Configuring VLAN Stacking	9-9
Configuring VLAN Stacking Services	9-11
Configuring SVLANs	9-12
Configuring a VLAN Stacking Service	9-13
Configuring VLAN Stacking Network Ports	9-14
Configuring NNI Port Parameters	9-14
Configuring a VLAN Stacking Service Access Point	9-15
Configuring VLAN Stacking User Ports	9-16
Configuring the Type of Customer Traffic to Tunnel	9-17
Configuring a Service Access Point Profile	9-18
Associating a Profile with a Service Access Point	9-20
Configuring a UNI Profile	9-20
Configuring Destination MAC Address	9-20
Associating UNI Profiles with UNI Ports	9-21
VLAN Stacking Application Examples	9-22
VLAN Stacking Configuration Example	9-23
Wire-Speed Ethernet Loopback Test	9-25
Configuring an Ethernet Loopback Test	9-25
Outward (Egress) Loopback Test	9-26
Inward (Ingress) Loopback Test	9-27
Verifying the VLAN Stacking Configuration	9-29
Chapter 10 Using 802.1Q 2005 Multiple Spanning Tree	10-1
In This Chapter	10-1
Spanning Tree Specifications	10-2
Spanning Tree Bridge Parameter Defaults	10-2
Spanning Tree Port Parameter Defaults	10-3
Multiple Spanning Tree Region Defaults	10-3
MST General Overview	10-4
How MSTP Works	10-4
Comparing MSTP with STP and RSTP	10-7
What is a Multiple Spanning Tree Instance (MSTI)	10-7
What is a Multiple Spanning Tree Region	10-8
What is the Common Spanning Tree	10-9
What is the Internal Spanning Tree (IST) Instance	10-9
What is the Common and Internal Spanning Tree Instance	10-9

MST Configuration Overview	10-10
Using Spanning Tree Configuration Commands	10-10
Understanding Spanning Tree Modes	10-11
MST Interoperability and Migration	10-12
Migrating from Flat Mode STP/RSTP to Flat Mode MSTP	10-12
Migrating from 1x1 Mode to Flat Mode MSTP	10-13
Quick Steps for Configuring an MST Region	10-14
Quick Steps for Configuring MSTIs	10-16
Verifying the MST Configuration	10-19

Chapter 11	Configuring Spanning Tree Parameters	11-1
	In This Chapter	11-2
	Spanning Tree Specifications	11-3
	Spanning Tree Bridge Parameter Defaults	11-4
	Spanning Tree Port Parameter Defaults	11-4
	Multiple Spanning Tree (MST) Region Defaults	11-5
	Ring Rapid Spanning Tree Defaults	11-5
	Spanning Tree Overview	11-6
	How the Spanning Tree Topology is Calculated	11-6
	Bridge Protocol Data Units (BPDU)	11-8
	Topology Examples	11-10
	Spanning Tree Operating Modes	11-12
	Using Flat Spanning Tree Mode	11-12
	Using 1x1 Spanning Tree Mode	11-13
	Using 1x1 Spanning Tree Mode with PVST+	11-14
	OmniSwitch PVST+ Interoperability	11-14
	BPDU Processing in PVST+ Mode	11-16
	Recommendations and Requirements for PVST+ Configurations	11-16
	Configuring STP Bridge Parameters	11-17
	Bridge Configuration Commands Overview	11-18
	Selecting the Bridge Protocol	11-20
	Configuring the Bridge Priority	11-20
	Configuring the Bridge Hello Time	11-21
	Configuring the Bridge Max Age Time	11-22
	Configuring the Bridge Forward Delay Time	11-23
	Enabling/Disabling the VLAN BPDU Switching Status	11-24
	Configuring the Path Cost Mode	11-24
	Using Automatic VLAN Containment	11-25
	Configuring STP Port Parameters	11-26
	Bridge Configuration Commands Overview	11-26
	Enabling/Disabling Spanning Tree on a Port	11-29
	Spanning Tree on Link Aggregate Ports	11-29
	Configuring Port Priority	11-30
	Port Priority on Link Aggregate Ports	11-31
	Configuring Port Path Cost	11-31

Path Cost for Link Aggregate Ports	11-32
Configuring Port Mode	11-34
Mode for Link Aggregate Ports	11-34
Configuring Port Connection Type	11-35
Connection Type on Link Aggregate Ports	11-36
Configuring Edge Port	11-36
Restricting Port Roles (Root Guard)	11-37
Restricting TCN Propagation	11-37
Limiting BPDU Transmission	11-37
Using RRSTP	11-38
Configuring RRSTP	11-39
Enabling and Disabling RRSTP	11-39
Creating and Removing RRSTP Rings	11-39
Sample Spanning Tree Configuration	11-40
Example Network Overview	11-40
Example Network Configuration Steps	11-41
Verifying the Spanning Tree Configuration	11-43
Chapter 12	
Configuring ERP	12-1
In This Chapter	12-1
ERP Specifications	12-2
ERP Defaults	12-2
ERP Overview	12-3
ERP Terms	12-3
ERP Timers	12-3
How Does ERP Work?	12-4
ERP Ring Modes	12-4
Overlapping Protected VLANs Between ERP Rings on same Node	12-6
ERP and RRSTP Differences	12-7
Interaction With Other Features	12-8
Spanning Tree	12-8
VLAN Stacking	12-8
Ethernet OAM	12-8
Quick Steps for Configuring ERP with Standard VLANs	12-9
Quick Steps for Configuring ERP with VLAN Stacking	12-10
ERP Configuration Overview and Guidelines	12-11
Configuring an ERP Ring	12-12
Adding Protected VLANs	12-13
Configuring an RPL Port	12-13
Setting the Wait-to-Restore Timer	12-14
Setting the Guard Timer	12-14
Monitoring Remote Ethernet OAM End Points with ERP	12-15
Configuring ERP with VLAN Stacking NNIs	12-16
Configuring ERP Protected SVLANs	12-17
Clearing ERP Statistics	12-18

	Sample Ethernet Ring Protection Configuration	12-19
	Example ERP Overview	12-19
	Example ERP Configuration Steps	12-20
	Verifying the ERP Configuration	12-21
Chapter 13	Configuring Loopback Detection	13-1
	In This Chapter	13-1
	LBD Specifications	13-1
	LBD Defaults	13-2
	Quick Steps for Configuring LBD	13-3
	LBD Overview	13-4
	Transmission Timer	13-4
	Autorecovery	13-4
	Interaction With Other Features	13-4
	Spanning Tree Protocol	13-4
	Link Aggregation	13-4
	Configuring LBD	13-5
	Enabling LBD	13-5
	Enabling LBD on a Port	13-5
	Configuring the LBD Transmission Timer	13-5
	Configuring the Autorecovery Timer	13-5
	Viewing LBD Statistics	13-5
	Recovering a Port from LBD Shutdown	13-6
	Verifying the LBD Configuration	13-7
Chapter 14	Configuring CPE Test Head	14-1
	In This Chapter	14-1
	CPE Test Head Specifications	14-2
	Quick Steps for Configuring CPE Test Head	14-3
	CPE Test Head Overview	14-5
	CPE Test Head Configuration Overview	14-6
	Configuration Guidelines	14-6
	Configuring a CPE Test Profile	14-7
	Running a CPE Test	14-9
	Stopping the CPE Test	14-9
	Verifying the CPE Test Configuration and Results	14-10
Chapter 15	Configuring Ethernet OAM	15-1
	In This Chapter	15-1
	Ethernet OAM Specifications	15-2
	Ethernet OAM Defaults	15-2
	Ethernet OAM Overview	15-3

Ethernet Service OAM	15-3
Elements of Service OAM	15-3
CFM Maintenance Domain	15-3
Fault Management	15-5
MIP CCM Database Support	15-5
Performance Monitoring	15-5
Interoperability with ITU-T Y.1731	15-7
Quick Steps for Configuring Service OAM	15-8
Configuring Ethernet OAM	15-9
Configuring a Maintenance Domain	15-9
Modifying a Maintenance Domain	15-10
Configuring a Maintenance Association	15-10
Configuring Maintenance Association Attributes	15-10
Configuring a Maintenance End Point	15-11
Configuring MEP Attributes	15-11
Configuring Loopback	15-11
Configuring Linktrace	15-12
Configuring the Fault Alarm Time	15-12
Configuring the Fault Reset Time	15-12
Configuring Ethernet Frame Delay Measurement	15-12
Verifying the Ethernet OAM Configuration	15-14
Chapter 16	
Service Assurance Agents (SAA)	16-1
In This Chapter	16-1
SAA Specifications	16-2
SAA Defaults	16-2
Quick Steps for Configuring SAA	16-3
Configuring Service Assurance Agent (SAA)	16-4
Configuring SAA for MAC Addresses	16-4
Configuring SAA for IP	16-4
Configuring SAA for Ethoam Loopback	16-4
Configuring SAA for ETH-DMM	16-4
Starting and Stopping SAAs	16-5
Displaying the SAA Configuration	16-5
Chapter 17	
Configuring EFM (LINK OAM)	17-1
In This Chapter	17-1
LINK OAM Specifications	17-2
LINK OAM Defaults	17-3
Quick Steps for Configuring LINK OAM	17-4
LINK OAM Overview	17-6
Discovery	17-6
Link Monitoring	17-7
Remote Fault detection	17-7
Remote Loopback Testing	17-8

	Interaction With Other Features	17-8
	Link Aggregate	17-8
	Connectivity Fault Management	17-8
	ERP	17-8
	Configuring LINK OAM	17-9
	Enabling and Disabling LINK OAM	17-9
	Setting the Transmit Delay	17-9
	Enabling and Disabling Propagation of Events	17-9
	Configuring Link Monitoring	17-10
	Enabling and Disabling Errored frame period	17-10
	Enabling and Disabling Errored frame	17-10
	Enabling and Disabling Errored frame seconds summary	17-10
	Configuring LINK OAM Loopback	17-11
	Enabling and Disabling Remote loopback	17-11
	Verifying the LINK OAM Configuration	17-12
Chapter 18	Configuring UDLD	18-1
	In This Chapter	18-1
	UDLD Specifications	18-2
	UDLD Defaults	18-2
	Quick Steps for Configuring UDLD	18-3
	UDLD Overview	18-4
	UDLD Operational Mode	18-4
	Normal Mode	18-4
	Aggressive Mode	18-4
	Mechanisms to Detect Unidirectional Links	18-5
	Neighbor database maintenance	18-5
	Echo detection	18-5
	Configuring UDLD	18-6
	Enabling and Disabling UDLD	18-6
	Enabling UDLD on a switch	18-6
	Disabling UDLD on a switch	18-6
	Enabling UDLD on a port	18-6
	Disabling UDLD on a port	18-6
	Configuring mode	18-7
	Configuring probe-timer	18-7
	Configuring echo-wait-timer	18-7
	Clearing UDLD Statistics	18-8
	Recovering a port from UDLD shutdown	18-8
	Displaying UDLD Information	18-8
Chapter 19	Configuring MAC Retention	19-1
	In This Chapter	19-1
	MAC Retention Defaults	19-2
	MAC Retention Overview	19-3

How MAC Retention Works	19-4
MAC Retention After Multiple Take-Overs	19-5
Configuring MAC Retention	19-6
Enabling MAC Retention	19-6
Detecting a Duplicate MAC Address	19-6
Configuring MAC Release	19-6
MAC Retention Applications	19-7
Software Failure	19-7
Link Failure	19-8
Chapter 20	
Configuring 802.1AB	20-1
In This Chapter	20-1
802.1AB Specifications	20-2
802.1AB Defaults Table	20-2
Quick Steps for Configuring 802.1AB	20-4
Quick Steps for Configuring LLDP-MED Network Policy	20-5
LLDP-MED Network Policy for Fixed Ports	20-5
LLDP on Mobile Ports	20-5
LLDP-MED Network Policy on 802.1x Ports	20-6
802.1AB Overview	20-8
LLDP-Media Endpoint Devices	20-9
LLDP-MED Network Policy	20-10
LLDP-MED Network Policy for VLAN Advertisement	20-10
Fast Restart of LLDP on Detection of MED	20-11
LLDP-MED for IP Phones	20-11
LLDP Agent Operation	20-11
LLDPDU Transmission and Reception	20-11
Aging Time	20-12
Nearest Bridge/Edge Mode	20-13
Nearest-Edge Mode Operation	20-13
Configuring 802.1AB	20-15
Configuring LLDPDU Flow	20-15
Enabling and Disabling Notification	20-15
Enabling and Disabling Management TLV	20-16
Enabling and Disabling 802.1 TLV	20-16
Enabling and Disabling 802.3 TLV	20-17
Enabling and Disabling MED TLV	20-17
Setting the Transmit Interval	20-18
Setting the Transmit Hold Multiplier Value	20-18
Setting the Transmit Delay	20-18
Setting the Transmit Fast Start Count	20-18
Setting the Reinit Delay	20-18
Setting the Notification Interval	20-18
Verifying 802.1AB Configuration	20-19

Chapter 21	Using Interswitch Protocols	21-1
	In This Chapter	21-1
	AIP Specifications	21-2
	AMAP Defaults	21-2
	AMAP Overview	21-3
	AMAP Transmission States	21-3
	Discovery Transmission State	21-4
	Common Transmission State	21-4
	Passive Reception State	21-4
	Common Transmission and Remote Switches	21-5
	Configuring AMAP	21-5
	Enabling or Disabling AMAP	21-5
	Configuring the AMAP Discovery Time-out Interval	21-5
	Configuring the AMAP Common Time-out Interval	21-6
	Displaying AMAP Information	21-7
Chapter 22	Configuring 802.1Q	22-1
	In this Chapter	22-1
	802.1Q Specifications	22-2
	802.1Q Defaults Table	22-2
	802.1Q Overview	22-3
	Configuring an 802.1Q VLAN	22-5
	Enabling Tagging on a Port	22-5
	Enabling Tagging with Link Aggregation	22-5
	Configuring the Frame Type	22-6
	Show 802.1Q Information	22-7
	Application Example	22-8
	Verifying 802.1Q Configuration	22-10
Chapter 23	Configuring Static Link Aggregation	23-1
	In This Chapter	23-1
	Static Link Aggregation Specifications	23-2
	Static Link Aggregation Default Values	23-2
	Quick Steps for Configuring Static Link Aggregation	23-3
	Static Link Aggregation Overview	23-5
	Static Link Aggregation Operation	23-5
	Relationship to Other Features	23-6
	Configuring Static Link Aggregation Groups	23-7
	Configuring Mandatory Static Link Aggregate Parameters	23-7
	Creating and Deleting a Static Link Aggregate Group	23-8
	Creating a Static Aggregate Group	23-8
	Deleting a Static Aggregate Group	23-8
	Adding and Deleting Ports in a Static Aggregate Group	23-9

	Adding Ports to a Static Aggregate Group	23-9
	Removing Ports from a Static Aggregate Group	23-9
	Modifying Static Aggregation Group Parameters	23-10
	Modifying the Static Aggregate Group Name	23-10
	Creating a Static Aggregate Group Name	23-10
	Deleting a Static Aggregate Group Name	23-10
	Modifying the Static Aggregate Group Administrative State	23-10
	Enabling the Static Aggregate Group Administrative State	23-10
	Disabling the Static Aggregate Group Administrative State	23-10
	Application Example	23-11
	Displaying Static Link Aggregation Configuration and Statistics	23-12
Chapter 24	Configuring Dynamic Link Aggregation	24-1
	In This Chapter	24-1
	Dynamic Link Aggregation Specifications	24-2
	Dynamic Link Aggregation Default Values	24-3
	Quick Steps for Configuring Dynamic Link Aggregation	24-4
	Dynamic Link Aggregation Overview	24-7
	Dynamic Link Aggregation Operation	24-7
	Relationship to Other Features	24-9
	Configuring Dynamic Link Aggregate Groups	24-10
	Configuring Mandatory Dynamic Link Aggregate Parameters	24-10
	Creating and Deleting a Dynamic Aggregate Group	24-11
	Creating a Dynamic Aggregate Group	24-11
	Deleting a Dynamic Aggregate Group	24-11
	Configuring Ports to Join and Removing Ports in a Dynamic Aggregate Group	24-12
	Configuring Ports To Join a Dynamic Aggregate Group	24-12
	Removing Ports from a Dynamic Aggregate Group	24-13
	Modifying Dynamic Link Aggregate Group Parameters	24-14
	Modifying Dynamic Aggregate Group Parameters	24-14
	Modifying the Dynamic Aggregate Group Name	24-14
	Modifying the Dynamic Aggregate Group Administrative State	24-15
	Configuring and Deleting the Dynamic Aggregate Group Actor Administrative Key	24-15
	Modifying the Dynamic Aggregate Group Actor System Priority	24-16
	Modifying the Dynamic Aggregate Group Actor System ID	24-16
	Modifying the Dynamic Aggregate Group Partner Administrative Key	24-17
	Modifying the Dynamic Aggregate Group Partner System Priority	24-17
	Modifying the Dynamic Aggregate Group Partner System ID	24-18
	Modifying Dynamic Link Aggregate Actor Port Parameters	24-18
	Modifying the Actor Port System Administrative State	24-19
	Modifying the Actor Port System ID	24-20
	Modifying the Actor Port System Priority	24-21
	Modifying the Actor Port Priority	24-22
	Modifying Dynamic Aggregate Partner Port Parameters	24-23
	Modifying the Partner Port System Administrative State	24-23

Modifying the Partner Port Administrative Key	24-25
Modifying the Partner Port System ID	24-25
Modifying the Partner Port System Priority	24-26
Modifying the Partner Port Administrative Status	24-27
Modifying the Partner Port Priority	24-27
Application Examples	24-29
Sample Network Overview	24-29
Link Aggregation and Spanning Tree Example	24-30
Link Aggregation and QoS Example	24-31
Displaying Dynamic Link Aggregation Configuration and Statistics	24-32

Chapter 25

Configuring IP	25-1
In This Chapter	25-1
IP Specifications	25-3
IP Defaults	25-3
Quick Steps for Configuring IP Forwarding	25-4
IP Overview	25-5
IP Protocols	25-5
Transport Protocols	25-5
Application-Layer Protocols	25-5
Additional IP Protocols	25-6
IP Forwarding	25-7
Configuring an IP Router Interface	25-8
Modifying an IP Router Interface	25-9
Removing an IP Router Interface	25-9
Configuring a Loopback0 Interface	25-10
Loopback0 Address Advertisement	25-10
Creating a Static Route	25-10
Creating a Default Route	25-11
Configuring Address Resolution Protocol (ARP)	25-12
Adding a Permanent Entry to the ARP Table	25-12
Deleting a Permanent Entry from the ARP Table	25-12
Clearing a Dynamic Entry from the ARP Table	25-13
Local Proxy ARP	25-13
ARP Filtering	25-14
IP Configuration	25-15
Configuring the DHCP Client Interface	25-15
Configuring the Router Primary Address	25-15
Configuring the Router ID	25-15
Configuring the Route Preference of a Router	25-15
Configuring the Time-to-Live (TTL) Value	25-16
Configuring Route Map Redistribution	25-16
Using Route Maps	25-16
Configuring Route Map Redistribution	25-21
Route Map Redistribution Example	25-22
IP-Directed Broadcasts	25-23
Denial of Service (DoS) Filtering	25-23

	Enabling/Disabling IP Services	25-28
	Managing IP	25-29
	Internet Control Message Protocol (ICMP)	25-29
	ICMP Control Table	25-32
	ICMP Statistics Table	25-32
	Using the Ping Command	25-32
	Tracing an IP Route	25-33
	Displaying TCP Information	25-33
	Displaying UDP Information	25-33
	Verifying the IP Configuration	25-34
Chapter 26	Configuring IPv6	26-1
	In This Chapter	26-1
	IPv6 Specifications	26-2
	IPv6 Defaults	26-3
	Quick Steps for Configuring IPv6 Routing	26-4
	IPv6 Overview	26-5
	IPv6 Addressing	26-6
	IPv6 Address Notation	26-7
	IPv6 Address Prefix Notation	26-7
	Autoconfiguration of IPv6 Addresses	26-8
	Configuring an IPv6 Interface	26-9
	Modifying an IPv6 Interface	26-9
	Removing an IPv6 Interface	26-10
	Assigning IPv6 Addresses	26-11
	Removing an IPv6 Address	26-12
	Creating an IPv6 Static Route	26-13
	Configuring the Route Preference of a Router	26-14
	Configuring Route Map Redistribution	26-15
	Using Route Maps	26-15
	Configuring Route Map Redistribution	26-19
	Route Map Redistribution Example	26-20
	Verifying the IPv6 Configuration	26-21
Chapter 27	Configuring RIP	27-1
	In This Chapter	27-1
	RIP Specifications	27-2
	RIP Defaults	27-2
	Quick Steps for Configuring RIP Routing	27-3
	RIP Overview	27-4
	RIP Version 2	27-5
	RIP Routing	27-6

Loading RIP	27-6
Enabling RIP	27-7
Creating a RIP Interface	27-7
Enabling a RIP Interface	27-7
Configuring the RIP Interface Send Option	27-7
Configuring the RIP Interface Receive Option	27-8
Configuring the RIP Interface Metric	27-8
Configuring the RIP Interface Route Tag	27-9
RIP Options	27-9
Configuring the RIP Forced Hold-Down Interval	27-9
Configuring the RIP Update Interval	27-9
Configuring the RIP Invalid Timer	27-10
Configuring the RIP Garbage Timer	27-10
Configuring the RIP Hold-Down Timer	27-10
Reducing the Frequency of RIP Routing Updates	27-10
Enabling a RIP Host Route	27-11
Configuring Redistribution	27-12
Using Route Maps	27-12
Configuring Route Map Redistribution	27-16
Route Map Redistribution Example	27-17
RIP Security	27-18
Configuring Authentication Type	27-18
Configuring Passwords	27-18
Verifying the RIP Configuration	27-19
Chapter 28	
Configuring RDP	28-1
In This Chapter	28-1
RDP Specifications	28-2
RDP Defaults	28-2
Quick Steps for Configuring RDP	28-3
RDP Overview	28-5
RDP Interfaces	28-6
Security Concerns	28-7
Enabling/Disabling RDP	28-8
Creating an RDP Interface	28-8
Specifying an Advertisement Destination Address	28-9
Defining the Advertisement Interval	28-9
Setting the Maximum Advertisement Interval	28-9
Setting the Minimum Advertisement Interval	28-10
Setting the Advertisement Lifetime	28-10
Setting the Preference Levels for Router IP Addresses	28-10
Verifying the RDP Configuration	28-11

Chapter 29	Configuring DHCP	29-1
	In This Chapter	29-1
	DHCP Relay Specifications	29-2
	DHCP Relay Defaults	29-3
	Quick Steps for Setting Up DHCP Relay	29-4
	DHCP Relay Overview	29-5
	DHCP	29-6
	DHCP and the OmniSwitch	29-6
	External DHCP Relay Application	29-6
	Internal DHCP Relay	29-8
	DHCP Relay Implementation	29-9
	Global DHCP	29-9
	Setting the IP Address	29-9
	Per-VLAN DHCP	29-9
	Identifying the VLAN	29-9
	Configuring BOOTP/DHCP Relay Parameters	29-10
	Setting the Forward Delay	29-10
	Setting Maximum Hops	29-11
	Setting the Relay Forwarding Option	29-11
	Configuring the DHCP Client Interface	29-12
	Configuring the DHCP Client Interface	29-12
	Reload and Takeover	29-12
	DHCP Client Interface Guidelines	29-13
	Configuring UDP Port Relay	29-14
	Enabling/Disabling UDP Port Relay	29-15
	Specifying a Forwarding VLAN	29-15
	Configuring DHCP Security Features	29-16
	Using the Relay Agent Information Option (Option-82)	29-16
	How the Relay Agent Processes DHCP Packets from the Client	29-17
	How the Relay Agent Processes DHCP Packets from the Server	29-17
	Enabling the Relay Agent Information Option-82	29-18
	Configuring a Relay Agent Information Option-82 Policy	29-18
	Using DHCP Snooping	29-19
	DHCP Snooping Configuration Guidelines	29-20
	Enabling DHCP Snooping	29-20
	Configuring the Port Trust Mode	29-22
	Bypassing the Option-82 Check on Untrusted Ports	29-22
	Configuring Port IP Source Filtering	29-23
	Configuring the DHCP Snooping Binding Table	29-23
	Layer 2 DHCP Snooping	29-25
	Verifying the DHCP Relay Configuration	29-26
Chapter 30	Configuring Access Guardian	30-1
	In This Chapter	30-1
	Access Guardian Specifications	30-2

Access Guardian Defaults	30-3
Quick Steps for Configuring Access Guardian	30-4
Quick Steps for Configuring User Network Profiles	30-6
Access Guardian Overview	30-7
Authentication and Classification	30-7
Using Device Classification Policies	30-8
User Network Profiles (Role-Based Access)	30-10
Interaction With Other Features	30-11
Captive Portal - Browser Support	30-11
Setting Up Port-Based Network Access Control	30-12
Setting 802.1X Switch Parameters	30-12
Enabling MAC Authentication	30-12
Enabling an Authentication Server Down Policy	30-12
Enabling 802.1X on Ports	30-13
Configuring 802.1X Port Parameters	30-13
Configuring Access Guardian Policies	30-14
Configuring Supplicant Policies	30-15
Supplicant Policy Examples	30-16
Configuring Non-supplicant Policies	30-17
Non-supplicant Policy Examples	30-18
Configuring the Captive Portal Policy	30-20
Configuring Captive Portal Authentication	30-22
Configuring Captive Portal Session Parameters	30-23
Customizing Captive Portal	30-23
Authenticating with Captive Portal	30-25
Logging Into the Network with Captive Portal	30-25
Logging Off the Network with Captive Portal	30-28
Configuring User Network Profiles	30-29
Verifying the Access Guardian Configuration	30-30
Chapter 31	
Managing Authentication Servers	31-1
In This Chapter	31-1
Authentication Server Specifications	31-2
Server Defaults	31-3
RADIUS Authentication Servers	31-3
TACACS+ Authentication Servers	31-3
LDAP Authentication Servers	31-3
Quick Steps For Configuring Authentication Servers	31-4
Server Overview	31-5
Backup Authentication Servers	31-5
Authenticated Switch Access	31-5
Port-Based Network Access Control (802.1X)	31-6
ACE/Server	31-7
Clearing an ACE/Server Secret	31-7

RADIUS Servers	31-8
RADIUS Server Attributes	31-8
Standard Attributes	31-8
Vendor-Specific Attributes for RADIUS	31-10
Configuring Functional Privileges on the Server	31-11
RADIUS Accounting Server Attributes	31-11
Configuring the RADIUS Client	31-12
TACACS+ Server	31-14
TACACS+ Client Limitations	31-14
Configuring the TACACS+ Client	31-15
LDAP Servers	31-16
Setting Up the LDAP Authentication Server	31-16
LDAP Server Details	31-17
LDIF File Structure	31-17
Common Entries	31-17
Directory Entries	31-18
Directory Searches	31-19
Retrieving Directory Search Results	31-19
Directory Modifications	31-19
Directory Compare and Sort	31-20
The LDAP URL	31-20
Password Policies and Directory Servers	31-21
Directory Server Schema for LDAP Authentication	31-22
Vendor-Specific Attributes for LDAP Servers	31-22
LDAP Accounting Attributes	31-23
Dynamic Logging	31-25
Configuring the LDAP Authentication Client	31-26
Creating an LDAP Authentication Server	31-27
Modifying an LDAP Authentication Server	31-27
Setting Up SSL for an LDAP Authentication Server	31-27
Removing an LDAP Authentication Server	31-28
Verifying the Authentication Server Configuration	31-28
Chapter 32 Configuring 802.1X	32-1
In This Chapter	32-1
802.1X Specifications	32-2
802.1X Defaults	32-2
Quick Steps for Configuring 802.1X	32-4
802.1X Overview	32-6
Supplicant Classification	32-6
802.1X Ports and DHCP	32-7
Re-authentication	32-7
802.1X Accounting	32-8
Setting Up Port-Based Network Access Control	32-9
Setting 802.1X Switch Parameters	32-9
Enabling MAC Authentication	32-9
Enabling 802.1X on Ports	32-9

	Configuring 802.1X Port Parameters	32-10
	Configuring the Port Control Direction	32-10
	Configuring the Port Authorization	32-10
	Configuring 802.1X Port Timeouts	32-10
	Configuring the Maximum Number of Requests	32-11
	Configuring the Number of Polling Retries	32-11
	Re-authenticating an 802.1X Port	32-11
	Initializing an 802.1X Port	32-12
	Configuring Accounting for 802.1X	32-12
	Verifying the 802.1X Port Configuration	32-13
Chapter 33	Managing Policy Servers	33-1
	In This Chapter	33-1
	Policy Server Specifications	33-2
	Policy Server Defaults	33-2
	Policy Server Overview	33-3
	Installing the LDAP Policy Server	33-3
	Modifying Policy Servers	33-4
	Modifying LDAP Policy Server Parameters	33-4
	Disabling the Policy Server From Downloading Policies	33-4
	Modifying the Port Number	33-5
	Modifying the Policy Server Username and Password	33-5
	Modifying the Searchbase	33-5
	Configuring a Secure Socket Layer for a Policy Server	33-6
	Loading Policies From an LDAP Server	33-6
	Removing LDAP Policies From the Switch	33-6
	Interaction With CLI Policies	33-7
	Verifying the Policy Server Configuration	33-7
Chapter 34	Configuring QoS	34-1
	In This Chapter	34-1
	QoS Specifications	34-2
	QoS General Overview	34-3
	QoS Policy Overview	34-4
	How Policies Are Used	34-4
	Valid Policies	34-5
	Policy Lists	34-5
	Interaction With Other Features	34-5
	Ethernet Service (VLAN Stacking)	34-6
	Condition Combinations	34-7
	Action Combinations	34-9
	Condition and Action Combinations	34-11

QoS Defaults	34-12
Global QoS Defaults	34-12
QoS Port Defaults	34-13
Policy Rule Defaults	34-13
Policy Action Defaults	34-14
Default (Built-in) Policies	34-14
QoS Configuration Overview	34-15
Configuring Global QoS Parameters	34-16
Enabling/Disabling QoS	34-16
Setting the Global Default Dispositions	34-16
Setting the Global Default Servicing Mode	34-17
Automatic QoS Prioritization	34-17
Configuring Automatic Prioritization for NMS Traffic	34-17
Configuring Automatic Prioritization for IP Phone Traffic	34-18
Using the QoS Log	34-18
What Kind of Information Is Logged	34-18
Number of Lines in the QoS Log	34-19
Log Detail Level	34-19
Forwarding Log Events	34-20
Forwarding Log Events to the Console	34-20
Displaying the QoS Log	34-20
Clearing the QoS Log	34-21
Classifying Bridged Traffic as Layer 3	34-21
Setting the Statistics Interval	34-22
Returning the Global Configuration to Defaults	34-22
Verifying Global Settings	34-22
QoS Ports and Queues	34-23
Shared Queues	34-23
Prioritizing and Queue Mapping	34-23
Maintaining the 802.1p Priority for IP Packets	34-24
Configuring Queuing Schemes	34-25
Configuring the Servicing Mode for a Port	34-26
Bandwidth Shaping	34-27
Configuring the Egress Queue Maximum Bandwidth	34-27
Setting the DEI Bit	34-27
Configuring the DEI Bit Setting	34-28
Trusted and Untrusted Ports	34-28
Configuring Trusted Ports	34-29
Using Trusted Ports With Policies	34-29
Verifying the QoS Port and Queue Configuration	34-30
Creating Policies	34-31
Quick Steps for Creating Policies	34-31
ASCII-File-Only Syntax	34-32
Creating Policy Conditions	34-33
Removing Condition Parameters	34-34
Deleting Policy Conditions	34-34
Creating Policy Actions	34-34
Removing Action Parameters	34-35
Deleting a Policy Action	34-35
Creating Policy Rules	34-35

Configuring a Rule Validity Period	34-36
Disabling Rules	34-37
Rule Precedence	34-37
Saving Rules	34-37
Logging Rules	34-38
Deleting Rules	34-38
Creating Policy Lists	34-38
Guidelines for Configuring Policy Lists	34-39
Using the Default Policy List	34-40
Using Egress Policy Lists	34-40
Policy List Examples	34-41
Verifying Policy Configuration	34-42
Testing Conditions	34-43
Using Condition Groups in Policies	34-46
ACLs	34-46
Sample Group Configuration	34-46
Creating Network Groups	34-47
Creating Services	34-48
Creating Service Groups	34-49
Creating MAC Groups	34-50
Creating Port Groups	34-51
Port Groups and Maximum Bandwidth	34-52
Creating VLAN Groups	34-53
Verifying Condition Group Configuration	34-55
Using Map Groups	34-56
Sample Map Group Configuration	34-56
How Map Groups Work	34-57
Creating Map Groups	34-57
Verifying Map Group Configuration	34-58
Applying the Configuration	34-59
Deleting the Pending Configuration	34-60
Flushing the Configuration	34-60
Interaction With LDAP Policies	34-61
Verifying the Applied Policy Configuration	34-61
Policy Applications	34-62
Basic QoS Policies	34-63
Basic Commands	34-63
Traffic Prioritization Example	34-63
Bandwidth Shaping Example	34-64
Tri-Color Marking	34-64
Configuring TCM Policies	34-65
Redirection Policies	34-67
Policy-Based Mirroring	34-67
ICMP Policy Example	34-68
802.1p and ToS/DSCP Marking and Mapping	34-68
Policy Based Routing	34-70

Chapter 35	Configuring ACLs	35-1
	In This Chapter	35-1
	ACL Specifications	35-2
	ACL Defaults	35-3
	Quick Steps for Creating ACLs	35-4
	ACL Overview	35-5
	Rule Precedence	35-6
	How Precedence is Determined	35-6
	Interaction With Other Features	35-6
	Valid Combinations	35-6
	ACL Configuration Overview	35-7
	Setting the Global Disposition	35-7
	Creating Condition Groups For ACLs	35-8
	Configuring ACLs	35-9
	Creating Policy Conditions For ACLs	35-9
	Creating Policy Actions For ACLs	35-10
	Creating Policy Rules for ACLs	35-10
	Layer 2 ACLs	35-10
	Layer 2 ACL Example	35-11
	Layer 3 ACLs	35-11
	Layer 3 ACL: Example 1	35-12
	Layer 3 ACL: Example 2	35-12
	IPv6 ACLs	35-13
	Multicast Filtering ACLs	35-13
	Using ACL Security Features	35-15
	Configuring a UserPorts Group	35-15
	Configuring UserPort Traffic Types and Port Behavior	35-16
	Configuring a DropServices Group	35-16
	Configuring ICMP Drop Rules	35-17
	Configuring TCP Connection Rules	35-17
	Verifying the ACL Configuration	35-19
	ACL Application Example	35-21
Chapter 36	Configuring IP Multicast Switching	36-1
	In This Chapter	36-1
	IPMS Specifications	36-3
	IPMSv6 Specifications	36-3
	IPMS Default Values	36-4
	IPMSv6 Default Values	36-5
	IPMS Overview	36-6
	IPMS Example	36-6
	Reserved IP Multicast Addresses	36-7

Configuring IPMS on a Switch	36-8
Enabling and Disabling IP Multicast Status	36-8
Enabling IP Multicast Status	36-8
Disabling IP Multicast Status	36-8
Enabling and Disabling IGMP Querier-forwarding	36-9
Enabling the IGMP Querier-forwarding	36-9
Disabling the IGMP Querier-forwarding	36-9
Configuring and Restoring the IGMP Version	36-9
Configuring the IGMP Version	36-10
Restoring the IGMP Version	36-10
Configuring and Removing an IGMP Static Neighbor	36-10
Configuring an IGMP Static Neighbor	36-10
Removing an IGMP Static Neighbor	36-11
Configuring and Removing an IGMP Static Querier	36-11
Configuring an IGMP Static Querier	36-11
Removing an IGMP Static Querier	36-11
Configuring and Removing an IGMP Static Group	36-11
Configuring an IGMP Static Group	36-12
Removing an IGMP Static Group	36-12
Modifying IPMS Parameters	36-13
Modifying the IGMP Query Interval	36-13
Configuring the IGMP Query Interval	36-13
Restoring the IGMP Query Interval	36-13
Modifying the IGMP Last Member Query Interval	36-13
Configuring the IGMP Last Member Query Interval	36-14
Restoring the IGMP Last Member Query Interval	36-14
Modifying the IGMP Query Response Interval	36-14
Configuring the IGMP Query Response Interval	36-14
Restoring the IGMP Query Response Interval	36-15
Modifying the IGMP Router Timeout	36-15
Configuring the IGMP Router Timeout	36-15
Restoring the IGMP Router Timeout	36-15
Modifying the Source Timeout	36-16
Configuring the Source Timeout	36-16
Restoring the Source Timeout	36-16
Enabling and Disabling IGMP Querying	36-17
Enabling the IGMP Querying	36-17
Disabling the IGMP Querying	36-17
Modifying the IGMP Robustness Variable	36-17
Configuring the IGMP Robustness variable	36-17
Restoring the IGMP Robustness Variable	36-18
Enabling and Disabling the IGMP Spoofing	36-18
Enabling the IGMP Spoofing	36-18
Disabling the IGMP Spoofing	36-18
Enabling and Disabling the IGMP Zapping	36-19
Enabling the IGMP Zapping	36-19
Disabling the IGMP Zapping	36-19
IPMSv6 Overview	36-20
IPMSv6 Example	36-20
Reserved IPv6 Multicast Addresses	36-21
MLD Version 2	36-21

Configuring IPMSv6 on a Switch	36-22
Enabling and Disabling IPv6 Multicast Status	36-22
Enabling IPv6 Multicast Status	36-22
Disabling IPv6 Multicast Status	36-22
Enabling and Disabling MLD Querier-forwarding	36-23
Enabling the MLD Querier-forwarding	36-23
Disabling the MLD Querier-forwarding	36-23
Configuring and Restoring the MLD Version	36-23
Configuring the MLD Version 2	36-23
Restoring the MLD Version 1	36-24
Configuring and Removing an MLD Static Neighbor	36-24
Configuring an MLD Static Neighbor	36-24
Removing an MLD Static Neighbor	36-25
Configuring and Removing an MLD Static Querier	36-25
Configuring an MLD Static Querier	36-25
Removing an MLD Static Querier	36-25
Configuring and Removing an MLD Static Group	36-25
Configuring an MLD Static Group	36-26
Removing an MLD Static Group	36-26
Modifying IPMSv6 Parameters	36-27
Modifying the MLD Query Interval	36-27
Configuring the MLD Query Interval	36-27
Restoring the MLD Query Interval	36-27
Modifying the MLD Last Member Query Interval	36-27
Configuring the MLD Last Member Query Interval	36-27
Restoring the MLD Last Member Query Interval	36-28
Modifying the MLD Query Response Interval	36-28
Configuring the MLD Query Response Interval	36-28
Restoring the MLD Query Response Interval	36-28
Modifying the MLD Router Timeout	36-29
Configuring the MLD Router Timeout	36-29
Restoring the MLD Router Timeout	36-29
Modifying the Source Timeout	36-29
Configuring the Source Timeout	36-30
Restoring the Source Timeout	36-30
Enabling and Disabling the MLD Querying	36-30
Enabling the MLD Querying	36-30
Disabling the MLD Querying	36-30
Modifying the MLD Robustness Variable	36-31
Configuring the MLD Robustness Variable	36-31
Restoring the MLD Robustness Variable	36-31
Enabling and Disabling the MLD Spoofing	36-32
Enabling the MLD Spoofing	36-32
Disabling the MLD Spoofing	36-32
Enabling and Disabling the MLD Zapping	36-32
Enabling the MLD Zapping	36-33
Disabling the MLD Zapping	36-33
IPMS Application Example	36-34
IPMSv6 Application Example	36-36

Displaying IPMS Configurations and Statistics	36-38
Displaying IPMSv6 Configurations and Statistics	36-39
Chapter 37	
Configuring IP Multicast VLAN	37-1
In This Chapter	37-1
IP Multicast VLAN Specifications	37-2
IP Multicast VLAN Defaults	37-2
IP Multicast VLAN Overview	37-3
VLAN Stacking Mode	37-3
IPMVLAN Lookup Mode	37-3
Enterprise Mode	37-4
IPMV Packet Flows	37-5
VLAN Stacking Mode	37-5
Enterprise Mode	37-8
Configuring IPMVLAN	37-9
Creating and Deleting IPMVLAN	37-9
Creating IPMVLAN	37-9
Deleting IPMVLAN	37-10
Assigning and Deleting IPv4/IPv6 Address	37-10
Assigning an IPv4/IPv6 Address to an IPMVLAN	37-10
Deleting an IPv4/IPv6 Address from an IPMVLAN	37-10
Assigning and Deleting a Customer VLAN Tag	37-10
Assigning C-Tag to an IPMVLAN	37-10
Deleting C-Tag from an IPMVLAN	37-10
Creating and Deleting a Sender Port	37-11
Creating a Sender Port in an IPMVLAN	37-11
Deleting a Sender Port from an IPMVLAN	37-11
Creating and Deleting a Receiver Port	37-11
Creating a Receiver Port in an IPMVLAN	37-11
Deleting a Receiver Port from an IPMVLAN	37-12
Associating an IPMVLAN with a Customer VLAN	37-12
IPMVLAN Application Example	37-13
Verifying the IP Multicast VLAN Configuration	37-15
Chapter 38	
Diagnosing Switch Problems	38-1
In This Chapter	38-1
Port Mirroring Overview	38-3
Port Mirroring Specifications	38-3
Port Mirroring Defaults	38-3
Quick Steps for Configuring Port Mirroring	38-4
Port Monitoring Overview	38-5
Port Monitoring Specifications	38-5
Port Monitoring Defaults	38-5
Quick Steps for Configuring Port Monitoring	38-6

sFlow Overview	38-7
sFlow Specifications	38-7
sFlow Defaults	38-7
Quick Steps for Configuring sFlow	38-8
Remote Monitoring (RMON) Overview	38-10
RMON Specifications	38-10
RMON Probe Defaults	38-11
Quick Steps for Enabling/Disabling RMON Probes	38-11
Switch Health Overview	38-12
Switch Health Specifications	38-12
Switch Health Defaults	38-13
Quick Steps for Configuring Switch Health	38-13
Port Mirroring	38-14
What Ports Can Be Mirrored?	38-14
How Port Mirroring Works	38-14
What Happens to the Mirroring Port	38-15
Mirroring on Multiple Ports	38-15
Using Port Mirroring with External RMON Probes	38-15
Remote Port Mirroring	38-17
Creating a Mirroring Session	38-18
Unblocking Ports (Protection from Spanning Tree)	38-19
Enabling or Disabling Mirroring Status	38-19
Disabling a Mirroring Session (Disabling Mirroring Status)	38-19
Configuring Port Mirroring Direction	38-20
Enabling or Disabling a Port Mirroring Session (Shorthand)	38-20
Displaying Port Mirroring Status	38-21
Deleting A Mirroring Session	38-21
Configuring Remote Port Mirroring	38-22
Port Monitoring	38-24
Configuring a Port Monitoring Session	38-25
Enabling a Port Monitoring Session	38-25
Disabling a Port Monitoring Session	38-25
Deleting a Port Monitoring Session	38-25
Pausing a Port Monitoring Session	38-26
Configuring Port Monitoring Session Persistence	38-26
Configuring a Port Monitoring Data File	38-26
Suppressing Port Monitoring File Creation	38-27
Configuring Port Monitoring Direction	38-27
Displaying Port Monitoring Status and Data	38-28
sFlow	38-29
sFlow Manager	38-29
Receiver	38-29
Sampler	38-30
Poller	38-30
Configuring a sFlow Session	38-30
Configuring a Fixed Primary Address	38-31
Displaying a sFlow Receiver	38-31
Displaying a sFlow Sampler	38-32
Displaying a sFlow Poller	38-32

Displaying a sFlow Agent	38-33
Deleting a sFlow Session	38-33
Remote Monitoring (RMON)	38-34
Ethernet Statistics	38-35
History (Control & Statistics)	38-35
Alarm	38-35
Event	38-35
Enabling or Disabling RMON Probes	38-36
Displaying RMON Tables	38-37
Displaying a List of RMON Probes	38-37
Displaying Statistics for a Particular RMON Probe	38-38
Sample Display for Ethernet Statistics Probe	38-38
Sample Display for History Probe	38-39
Sample Display for Alarm Probe	38-39
Displaying a List of RMON Events	38-40
Displaying a Specific RMON Event	38-40
Monitoring Switch Health	38-41
Configuring Resource and Temperature Thresholds	38-43
Displaying Health Threshold Limits	38-44
Configuring Sampling Intervals	38-45
Viewing Sampling Intervals	38-45
Viewing Health Statistics for the Switch	38-46
Viewing Health Statistics for a Specific Interface	38-47
Resetting Health Statistics for the Switch	38-47

Chapter 39	Using Switch Logging	39-1
	In This Chapter	39-1
	Switch Logging Specifications	39-2
	Switch Logging Defaults	39-3
	Quick Steps for Configuring Switch Logging	39-4
	Switch Logging Overview	39-5
	Switch Logging Commands Overview	39-6
	Enabling Switch Logging	39-6
	Setting the Switch Logging Severity Level	39-6
	Specifying the Severity Level	39-8
	Removing the Severity Level	39-9
	Specifying the Switch Logging Output Device	39-9
	Enabling/Disabling Switch Logging Output to the Console	39-9
	Enabling/Disabling Switch Logging Output to Flash Memory	39-9
	Specifying an IP Address for Switch Logging Output	39-9
	Disabling an IP Address from Receiving Switch Logging Output	39-10
	Displaying Switch Logging Status	39-10
	Configuring the Switch Logging File Size	39-11
	Clearing the Switch Logging Files	39-11
	Displaying Switch Logging Records	39-12

Appendix A	Software License and Copyright Statements	A-1
	Alcatel-Lucent License Agreement	A-1
	ALCATEL-LUCENT SOFTWARE LICENSE AGREEMENT	A-1
	Third Party Licenses and Notices	A-4
	A. Booting and Debugging Non-Proprietary Software	A-4
	B. The OpenLDAP Public License: Version 2.8, 17 August 2003	A-4
	C. Linux	A-5
	D. GNU GENERAL PUBLIC LICENSE: Version 2, June 1991	A-5
	E. University of California	A-10
	F. Carnegie-Mellon University	A-10
	G. Random.c	A-10
	H. Apptitude, Inc.	A-11
	I. Agranat	A-11
	J. RSA Security Inc.	A-11
	K. Sun Microsystems, Inc.	A-12
	L. Wind River Systems, Inc.	A-12
	M. Network Time Protocol Version 4	A-12
	N. Remote-ni	A-13
	O. GNU Zip	A-13
	P. FREESCALE SEMICONDUCTOR SOFTWARE LICENSE AGREEMENT	A-13
	Q. Boost C++ Libraries	A-14
	R. U-Boot	A-14
	S. Solaris	A-14
	T. Internet Protocol Version 6	A-14
	U. CURSES	A-15
	V. ZModem	A-15
	W. Boost Software License	A-15
	X. OpenLDAP	A-15
	Y. BITMAP.C	A-16
	Z. University of Toronto	A-16
	AA.Free/OpenBSD	A-16
	Index	Index-1

About This Guide

This *OmniSwitch 6250 Network Configuration Guide* describes how to set up and monitor software features that will allow your switch to operate in a live network environment. The software features described in this manual are shipped standard with your OmniSwitch 6250 Series switches. These features are used when setting up your OmniSwitch in a network of switches and routers.

Supported Platforms

The information in this guide applies to the following products:

- Omniswitch 6250-Enterprise Models
- OmniSwitch 6250-Metro Models

Note. This *OmniSwitch 6250 Network Configuration Guide* covers Release 6.6.2, which is supported on the OmniSwitch 6250 Series.

Unsupported Platforms

The information in this guide does not apply to the following products:

- OmniSwitch 9000 Series
- OmniSwitch 6400 Series
- OmniSwitch 6600 Family
- OmniSwitch 6800 Family
- OmniSwitch 6850 Series
- OmniSwitch 6855 Series
- OmniSwitch (original version with no numeric model name)
- OmniSwitch 7700/7800
- OmniSwitch 8800
- Omni Switch/Router
- OmniStack
- OmniAccess

Who Should Read this Manual?

The audience for this user guide is network administrators and IT support personnel who need to configure, maintain, and monitor switches and routers in a live network. However, anyone wishing to gain knowledge on how fundamental software features are implemented in the OmniSwitch 6250 Series will benefit from the material in this configuration guide.

When Should I Read this Manual?

Read this guide as soon as you are ready to integrate your OmniSwitch into your network and you are ready to set up advanced routing protocols. You should already be familiar with the basics of managing a single OmniSwitch as described in the *OmniSwitch 6250 Switch Management Guide*.

The topics and procedures in this manual assume an understanding of the OmniSwitch stacking, directory structure, and basic switch administration commands and procedures. This manual will help you set up your switches to communicate with other switches in the network. The topics in this guide include VLANs, authentication, and Quality of Service (QoS)—features that are typically deployed in a multi-switch environment.

What is in this Manual?

This configuration guide includes information about configuring the following features:

- VLANs, VLAN router ports, mobile ports, and VLAN rules.
- Basic Layer 2 functions, such as Ethernet port parameters, source learning, Spanning Tree, and Alcatel interswitch protocols (AMAP and GMAP).
- Advanced Layer 2 functions, such as 802.1Q tagging, Link Aggregation, and IP Multicast Switching.
- Basic routing protocols and functions, such as static IP routes, RIP, and DHCP Relay.
- Security features, such as switch access control, authentication servers, and policy management.
- Quality of Service (QoS) and Access Control Lists (ACLs) features, such as policy rules for prioritizing and filtering traffic, and remapping packet headers.
- Diagnostic tools, such as RMON, port mirroring, and switch logging.

What is Not in this Manual?

The configuration procedures in this manual use Command Line Interface (CLI) commands in all examples. CLI commands are text-based commands used to manage the switch through serial (console port) connections or via Telnet sessions. Procedures for other switch management methods, such as web-based (WebView or OmniVista) or SNMP, are outside the scope of this guide.

For information on WebView and SNMP switch management methods consult the *OmniSwitch 6250 Switch Management Guide*. Information on using WebView and OmniVista can be found in the context-sensitive on-line help available with those network management applications.

This guide provides overview material on software features, how-to procedures, and application examples that will enable you to begin configuring your OmniSwitch. It is not intended as a comprehensive reference to all CLI commands available in the OmniSwitch. For such a reference to all OmniSwitch 6250 CLI commands, consult the *OmniSwitch 6250 CLI Reference Guide*.

How is the Information Organized?

Chapters in this guide are broken down by software feature. The titles of each chapter include protocol or features names (e.g., 802.1Q) with which most network professionals will be familiar.

Each software feature chapter includes sections that will satisfy the information requirements of casual readers, rushed readers, serious detail-oriented readers, advanced users, and beginning users.

Quick Information. Most chapters include a *specifications table* that lists RFCs and IEEE specifications supported by the software feature. In addition, this table includes other pertinent information such as minimum and maximum values and sub-feature support. Most chapters also include a *defaults table* that lists the default values for important parameters along with the CLI command used to configure the parameter. Many chapters include a *Quick Steps* section, which is a procedure covering the basic steps required to get a software feature up and running.

In-Depth Information. All chapters include *overview sections* on the software feature as well as on selected topics of that software feature. *Topical sections* may often lead into *procedure sections* that describe how to configure the feature just described. Serious readers and advanced users will also find the many *application examples*, located near the end of chapters, helpful. Application examples include diagrams of real networks and then provide solutions using the CLI to configure a particular feature, or more than one feature, within the illustrated network.

Documentation Roadmap

The OmniSwitch user documentation suite was designed to supply you with information at several critical junctures of the configuration process. The following section outlines a roadmap of the manuals that will help you at each stage of the configuration process. Under each stage, we point you to the manual or manuals that will be most helpful to you.

Stage 1: Using the Switch for the First Time

Pertinent Documentation: *Getting Started Guide*
Release Notes

A hard-copy *Getting Started Guide* is included with your switch; this guide provides all the information you need to get your switch up and running the first time. It provides information on unpacking the switch, rack mounting the switch, installing NI modules, unlocking access control, setting the switch's IP address, and setting up a password. It also includes succinct overview information on fundamental aspects of the switch, such as hardware LEDs, the software directory structure, CLI conventions, and web-based management.

At this time you should also familiarize yourself with the Release Notes that accompanied your switch. This document includes important information on feature limitations that are not included in other user guides.

Stage 2: Gaining Familiarity with Basic Switch Functions

Pertinent Documentation: *Hardware Users Guide*
Switch Management Guide

Once you have your switch up and running, you will want to begin investigating basic aspects of its hardware and software. Information about switch hardware is provided in the *Hardware Users Guide*. This guide provides specifications, illustrations, and descriptions of all hardware components, such as chassis, power supplies, Chassis Management Modules (CMMs), Network Interface (NI) modules, and cooling fans. It also includes steps for common procedures, such as removing and installing switch components.

The *Switch Management Guide* is the primary users guide for the basic software features on a single switch. This guide contains information on the switch directory structure, basic file and directory utilities, switch access security, SNMP, and web-based management. It is recommended that you read this guide before connecting your switch to the network.

Stage 3: Integrating the Switch Into a Network

Pertinent Documentation: *Network Configuration Guide*

When you are ready to connect your switch to the network, you will need to learn how the OmniSwitch implements fundamental software features, such as 802.1Q, VLANs, Spanning Tree, and network routing protocols. The *Network Configuration Guide* contains overview information, procedures, and examples on how standard networking technologies are configured in the OmniSwitch.

Anytime

The *OmniSwitch CLI Reference Guide* contains comprehensive information on all CLI commands supported by the switch. This guide includes syntax, default, usage, example, related CLI command, and CLI-to-MIB variable mapping information for all CLI commands supported by the switch. This guide can be consulted anytime during the configuration process to find detailed and specific information on each CLI command.

Related Documentation

The following are the titles and descriptions of all the related OmniSwitch 6250 user manuals:

- *OmniSwitch 6250 Series Getting Started Guide*

Describes the hardware and software procedures for getting an OmniSwitch 6250 Series switch up and running. Also provides information on fundamental aspects of OmniSwitch software and stacking architecture.

- *OmniSwitch 6250 Series Hardware Users Guide*

Complete technical specifications and procedures for all OmniSwitch 6250 Series chassis, power supplies, and fans. Also includes comprehensive information on assembling and managing stacked configurations.

- *OmniSwitch CLI Reference Guide*

Complete reference to all CLI commands supported on the OmniSwitch 6250. Includes syntax definitions, default values, examples, usage guidelines and CLI-to-MIB variable mappings.

- *OmniSwitch 6250 Switch Management Guide*

Includes procedures for readying an individual switch for integration into a network. Topics include the software directory architecture, image rollback protections, authenticated switch access, managing switch files, system configuration, using SNMP, and using web management software (WebView).

- *OmniSwitch 6250 Network Configuration Guide*

Includes network configuration procedures and descriptive information on all the major software features and protocols included in the base software package. Chapters cover Layer 2 information (Ethernet and VLAN configuration), Layer 3 information (routing protocols, such as RIP), security options (authenticated VLANs), Quality of Service (QoS), and link aggregation.

- *OmniSwitch Transceivers Guide*

Includes information on Small Form Factor Pluggable (SFPs) and 10 Gbps Small Form Factor Pluggables (XFPs) transceivers.

- *Technical Tips, Field Notices*

Includes information published by Alcatel's Customer Support group.

- *Release Notes*

Includes critical Open Problem Reports, feature exceptions, and other important information on the features supported in the current release and any limitations to their support.

User Manual CD

Some products are shipped with documentation included on a User Manual CD that accompanies the switch. This CD also includes documentation for other Alcatel data enterprise products.

All products are shipped with a Product Documentation Card that provides details for downloading documentation for all OmniSwitch and other Alcatel-Lucent data enterprise products.

All documentation is in PDF format and requires the Adobe Acrobat Reader program for viewing. Acrobat Reader freeware is available at www.adobe.com.

Note. In order to take advantage of the documentation CD's global search feature, it is recommended that you select the option for *searching PDF files* before downloading Acrobat Reader freeware.

To verify that you are using Acrobat Reader with the global search option, look for the following button in the toolbar:



Note. When printing pages from the documentation PDFs, de-select Fit to Page if it is selected in your print dialog. Otherwise pages may print with slightly smaller margins.

Technical Support

An Alcatel service agreement brings your company the assurance of 7x24 no-excuses technical support. You'll also receive regular software updates to maintain and maximize your Alcatel product's features and functionality and on-site hardware replacement through our global network of highly qualified service delivery partners. Additionally, with 24-hour-a-day access to Alcatel's Service and Support web page, you'll be able to view and update any case (open or closed) that you have reported to Alcatel's technical support, open a new case or access helpful release notes, technical bulletins, and manuals. For more information on Alcatel's Service Programs, see our web page at service.esd.alcatel-lucent.com, call us at 1-800-995-2696, or email us at support@ind.alcatel.com.

1 Configuring Ethernet Ports

The Ethernet software is responsible for a variety of functions that support Ethernet and Gigabit Ethernet, ports on OmniSwitch Series switches. These functions include diagnostics, software loading, initialization, configuration of line parameters, gathering statistics, and responding to administrative requests from SNMP or CLI.

In This Chapter

This chapter describes your switch's Ethernet port parameters and how to configure them through the Command Line Interface (CLI). CLI Commands are used in the configuration examples. For more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- [“Setting Ethernet Parameters for All Port Types” on page 1-7](#)
- [“Setting Ethernet Parameters for Non-Combo Ports” on page 1-12](#)
- [“Setting Ethernet Combo Port Parameters” on page 1-17](#)
- [“Verifying Ethernet Port Configuration” on page 1-22](#)

For information about CLI commands that can be used to view Ethernet port parameters, see the *OmniSwitch 6250 CLI Reference Guide*.

Ethernet Specifications

IEEE Standards Supported	802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) 802.3u (100BaseTX) 802.3ab (1000BaseT) 802.3z (1000Base-X)
Platforms Supported	OmniSwitch 6250 Series
Ports Supported	Ethernet (10 Mbps) Fast Ethernet (100 Mbps) Gigabit Ethernet (1 Gb/1000 Mbps)
Switching/Routing Support	Layer 2 Switching/Layer 3 Routing
Backbone Support	Fast Ethernet, Gigabit Ethernet
Port Mirroring Support	Fast Ethernet and Gigabit Ethernet ports
802.1Q Hardware Tagging	Fast Ethernet, Gigabit Ethernet
Jumbo Frame Configuration	Supported on Gigabit Ethernet
Maximum Frame Size	1553 bytes (10/100 Mbps) 9216 bytes (1 Gbps)

Ethernet Port Defaults (All Port Types)

The following table shows Ethernet port default values:

Parameter Description	Command	Default Value/Comments
Trap Port Link Messages	trap port link	Disabled
Interface Configuration	interfaces admin	Up (Enabled)
Flood Only Rate Limiting	interfaces flood rate	Enable
Multicast Rate Limiting	interfaces flood multicast	Disable
Peak Flood Rate Configuration	interfaces flood rate	4 Mbps (10 Ethernet) 49 Mbps (100 Fast Ethernet) 496 Mbps (1 Gigabit Ethernet)
Interface Alias	interfaces alias	None configured
Inter-Frame Gap	interfaces ifg	12 bytes
Maximum Frame Size	interfaces max frame	1553 (untagged) Ethernet packets 1553 (tagged) Ethernet packets 9216 Gigabit Ethernet packets

Non-Combo Port Defaults

The following table shows non-combo port default values:

Parameter Description	Command	Default Value/Comments
Interface Line Speed	interfaces speed	Auto (copper ports) 100 Mbps (fiber ports) 1 Gbps (GNI ports)
Duplex Mode	interfaces duplex	Auto (copper ports)/Full (fiber, GNI and XNI ports)
Autonegotiation	interfaces autoneg	Enable for all copper ports; Disable for all fiber ports
Crossover	interfaces crossover	Auto for all copper ports; MDI for all fiber ports (not configurable on fiber ports)
Flow Control (pause)	interfaces pause	Disabled

Combo Ethernet Port Defaults

The following table shows combo Ethernet port default values:

Parameter Description	Command	Default Value/Comments
Interface Line Speed	interfaces hybrid speed	Auto
Duplex Mode	interfaces hybrid duplex	Auto
Autonegotiation	interfaces hybrid autoneg	Enable
Crossover	interfaces hybrid crossover	Auto for all copper ports
Flow Control (pause)	interfaces hybrid pause	Disabled

Ethernet Ports Overview

This chapter describes the Ethernet software CLI commands used for configuring and monitoring your switch's Ethernet port parameters. These commands allow you to handle administrative or port-related requests to and from SNMP, CLI, or WebView.

OmniSwitch Series Combo Ports

The OmniSwitch platforms mentioned above have ports that are shared between copper 10/100/1000 RJ-45 connections and SFP connectors, which can accept any qualified SFP transceivers. These ports are known as *combo* ports (also sometimes referred to as “hybrid” ports).

You can use either the copper 10/100/1000 port or the equivalent SFP connector, for example, but not both at the same time. **By default, the switch will use the SFP connector instead of the equivalent copper RJ-45 port.** However, if the SFP connector goes down, the equivalent combo port will come up. This can be used if you want to use the SFP connector as your main link while having a copper link as a backup.

Note. See [“Valid Port Settings on OmniSwitch 6250 Series Switches” on page 1-5](#) for more information on combo ports. In addition, refer to the specific Hardware Users Guide for each type of switch.

See [“Setting Interface Line Speed for Combo Ports” on page 1-17](#) for more information on configuring combo ports.

Note: Settings for SFPs are dependent upon the type of transceiver being used. Refer to the OmniSwitch Transceivers Guide for information on supported SFPs.

Valid Port Settings on OmniSwitch 6250 Series Switches

This table below lists valid speed, duplex, and autonegotiation settings for the different OmniSwitch 6250 Series port types.

Chassis Type (Port Nos.)	Port Type	User-Specified Port Speed (Mbps) Supported	User-Specified Duplex Supported	Auto Negotiation Supported?
OmniSwitch 6250 Non-combo ports	RJ-45	auto/10/100/	auto/full/half	Yes
OmniSwitch 6250 Combo ports	RJ-45/SFP	RJ-45: auto/10/100/1000 SFP: Dependent	RJ-45: auto/full/half SFP: Dependent	RJ-45: Yes SFP: Dependent

See the *OmniSwitch 6250 Series Hardware Users Guide* for more information about the OmniSwitch 6250 hardware.

10/100/1000 Crossover Supported

By default, automatic crossover between MDI/MDIX (Media Dependent Interface/Media Dependent Interface with Crossover) media is supported on all the OmniSwitch ports. Therefore, either straight-through or crossover cable can be used between two ports as long as autonegotiation is configured on both sides of the link. See [“Configuring Autonegotiation and Crossover Settings” on page 1-14](#) for more information.

Autonegotiation Guidelines

Please note a link will not be established on any copper Ethernet port if any one of the following is true:

- The local port advertises 100 Mbps full duplex and the remote link partner is forced to 100 Mbps full duplex.
- The local port advertises 100 Mbps full duplex and the remote link partner is forced to 100 Mbps half duplex.
- The local port advertises 10 Mbps full duplex and the remote link partner is forced to 10 Mbps full duplex.
- The local port advertises 10 Mbps full duplex and the remote link partner is forced to 10 half duplex.

This is due to the fact that when the local device is set to auto negotiating 10/100 full duplex it senses the remote device is not auto negotiating. Therefore it resolves to Parallel Detect with Highest Common Denominator (HCD), which is “10/100 Half” according to IEEE 802.3 Clause 28.2.3.1.

However, since the local device is set to auto negotiating at 10/100 full duplex it cannot form a 10/100 Mbps half duplex link in any of the above mentioned cases. One solution is to configure the local device to autonegotiation, 10/100 Mbps, with auto or half duplex.

Flow Control and Autonegotiation

PAUSE frames are used to pause the flow of traffic between two connected devices when traffic congestion occurs. Flow control provides the ability to configure whether or not the switch will honor or transmit and honor PAUSE frames on an active interface. This feature is only supported on switch interfaces configured to run in full-duplex mode.

In addition to configuring flow control settings, this feature also works in conjunction with autonegotiation to determine operational transmit/receive settings for PAUSE frames between two switches. Note that the operational settings, as shown in the following table, override the configured settings as long as autonegotiation and flow control are both enabled for the interface:

Configured Local Tx	Configured Local Rx	Configured Remote Tx	Configured Remote Rx	Operational Local Tx	Operational Local Rx
No	No	No	No	No	No
Yes	Yes	Yes	Yes	Yes	Yes
Yes	No	Yes	No	No	No
No	Yes	No	Yes	Yes	Yes
No	No	No	Yes	No	No
Yes	Yes	No	No	No	No
Yes	No	Yes	Yes	No	No
No	Yes	Yes	No	No	Yes
No	No	Yes	No	No	No
Yes	Yes	No	Yes	Yes	Yes
Yes	No	No	No	No	No
No	Yes	Yes	Yes	Yes	Yes
No	No	Yes	Yes	No	No
Yes	Yes	Yes	No	No	No
Yes	No	No	Yes	Yes	No
No	Yes	No	No	No	No

If autonegotiation is disabled, the configured flow control settings are applied to the local interface. See [“Configuring Flow Control on Non-Combo Ports”](#) on page 1-15 and [“Configuring Flow Control on Combo Ports”](#) on page 1-20 for more information.

Setting Ethernet Parameters for All Port Types

The following sections describe how to configure Ethernet port parameters using CLI commands that can be used on all port types. See [“Setting Ethernet Parameters for Non-Combo Ports”](#) on page 1-12 for information on configuring non-combo ports and see [“Setting Ethernet Combo Port Parameters”](#) on page 1-17 for more information on configuring combo ports.

Setting Trap Port Link Messages

The **trap port link** command can be used to enable or disable (the default) trap port link messages on a specific port, a range of ports, or all ports on a switch (slot). When enabled, a trap message will be displayed on a Network Management Station (NMS) whenever the port state has changed.

Enabling Trap Port Link Messages

To enable trap port link messages on an entire switch, enter **trap** followed by the slot number and **port link enable**. For example, to enable trap port link messages on all ports on slot 2, enter:

```
-> trap 2 port link enable
```

To enable trap port link messages on a single port, enter **trap** followed by the slot number, a slash (/), the port number, and **port link enable**. For example, to enable trap port link messages on slot 2 port 3, enter:

```
-> trap 2/3 port link enable
```

To enable trap port link messages on a range of ports, enter **trap** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, and **port link enable**. For example, to enable trap port link messages ports 3 through 5 on slot 2, enter:

```
-> trap 2/3-5 port link enable
```

Disabling Trap Port Link Messages

To disable trap port link messages on an entire switch, enter **trap** followed by the slot number and **port link disable**. For example, to disable trap port link messages on all ports on slot 2, enter:

```
-> trap 2 port link disable
```

To disable trap port link messages on a single port, enter **trap** followed by the slot number, a slash (/), the port number, and **port link disable**. For example, to disable trap port link messages on slot 2 port 3, enter:

```
-> trap 2/3 port link disable
```

To disable trap port link messages on a range of ports, enter **trap** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, and **port link disable**. For example, to disable trap port link messages ports 3 through 5 on slot 2, enter:

```
-> trap 2/3-5 port link disable
```

Resetting Statistics Counters

The **interfaces no l2 statistics** command is used to reset all Layer 2 statistics counters on a specific port, a range of ports, or all ports on a switch (slot).

To reset Layer 2 statistics on an entire slot, enter **interfaces** followed by the slot number and **no l2 statistics**. For example, to reset all Layer 2 statistics counters on slot 2, enter:

```
-> interfaces 2 no l2 statistics
```

To reset Layer 2 statistics on a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, and **no l2 statistics**. For example, to reset all Layer 2 statistics counters on port 3 on slot 2, enter:

```
-> interfaces 2/3 no l2 statistics
```

To reset Layer 2 statistics on a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, and **no l2 statistics**. For example, to reset all Layer 2 statistics counters on ports 1 through 3 on slot 2, enter:

```
-> interfaces 2/1-3 no l2 statistics
```

Note. The **show interfaces**, **show interfaces accounting**, and **show interfaces counters** commands can be used to display Layer 2 statistics (e.g., input and output errors, deferred frames received, unicast packets transmitted). For information on using these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Enabling and Disabling Interfaces

The **interfaces admin** command is used to enable (the default) or disable a specific port, a range of ports, or all ports on an entire switch (NI module).

To enable or disable an entire slot, enter **interfaces** followed by the slot number, **admin**, and the desired administrative setting (either **up** or **down**). For example, to administratively disable slot 2, enter:

```
-> interfaces 2 admin down
```

To enable or disable a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, **admin**, and the desired administrative setting (either **up** or **down**). For example, to administratively disable port 3 on slot 2, enter:

```
-> interfaces 2/3 admin down
```

To enable or disable a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **admin**, and the desired administrative setting (either **up** or **down**). For example, to administratively disable ports 1 through 3 on slot 2, enter:

```
-> interfaces 2/1-3 admin down
```

Configuring Flood Rate Limiting

The following subsections describe how to apply a peak flood rate value to limit flooded traffic (see [“Flood Only Rate Limiting” on page 1-9](#)), limit multicast traffic (see [“Multicast Flood Rate Limiting” on page 1-9](#)), and configure the flood rate value for an entire switch (slot), a specific port, or a range of ports (see [“Configuring the Peak Flood Rate Value” on page 1-10](#)).

Flood Only Rate Limiting

The peak flood rate value is always applied to flooded traffic. However, it is also possible to apply this value to limit the rate of multicast traffic on any given port (see [“Multicast Flood Rate Limiting” on page 1-9](#)). The **interfaces flood rate** command automatically disables any multicast flood rate limiting on a port so that the peak flood rate is only applied to flooded traffic.

Note. The **interfaces flood multicast** command can also disable multicast flood rate limiting and is available on all the OmniSwitch Series switches.

To specify flood only rate limiting for a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, and **flood**. For example, the following command applies flood only rate limiting to port 2/3:

```
-> interfaces 2/3 flood
```

To specify flood only rate limiting for a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, and **flood**. For example, the following command applies flood only rate limiting to ports 3 through 4 on slot 2:

```
-> interfaces 2/3-4 flood
```

To configure the peak rate value used for flood only rate limiting, see [“Configuring the Peak Flood Rate Value” on page 1-10](#) for more information.

Multicast Flood Rate Limiting

The **interfaces flood multicast** command is used to enable or disable flood rate limiting for multicast traffic on a single port, a range of ports, or all ports on a switch (slot). When multicast flood rate limiting is enabled, the peak flood rate value for a port is applied to both multicast and flooded traffic.

By default, multicast flood rate limiting is disabled for a port. To apply the peak flood rate value to multicast traffic on a slot, enter **interfaces** followed by the slot number and **flood multicast**. For example, to enable the maximum flood rate for multicast traffic on slot 2, enter:

```
-> interfaces 2 flood multicast
```

To apply the peak flood rate value to multicast traffic on a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, and **flood multicast**. For example, to enable the maximum flood rate for multicast traffic on port 3 on slot 2, enter:

```
-> interfaces 2/3 flood multicast
```

To apply the peak flood rate value to multicast traffic on a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, and **flood multicast**. For example, to enable the maximum flood rate for multicast traffic on ports 3 through 4 on slot 2, enter:

```
-> interfaces 2/3-4 flood multicast
```

Note. Enabling multicast flood rate limiting with the **interfaces flood multicast** command will limit IP Multicast Switching (IPMS) and non-IPMS multicast traffic.

Configuring the Peak Flood Rate Value

The **interfaces flood rate** command is used to configure the peak flood rate value on a specific port, a range of ports, or all ports on a switch (slot) in megabits per second. Note the following regarding the configuration of this value:

- The **interfaces flood rate** command configures a maximum *ingress* flood rate value for an interface. This peak flood rate value is applied to flooded (unknown destination address, broadcast) and multi-cast traffic combined. For example, if an interface is configured with a peak flood rate of 500 Mbps, the 500 Mbps limit is shared by all traffic types.
- Although you can configure a flood rate equal to the line speed you should not do so. Alcatel-Lucent recommends that you always configure the flood rate to be less than the line speed.

By default the following peak flood rate values are used for limiting the rate at which traffic is flooded on a switch port:

parameter	default
<i>Mbps</i> (10 Ethernet)	4
<i>Mbps</i> (100 Fast Ethernet)	49
<i>Mbps</i> (Gigabit Ethernet)	496

To change the peak flood rate for an entire slot, enter **interfaces** followed by the slot number, **flood rate**, and the flood rate in megabits. For example, to configure the peak flood rate on slot 2 as 49 megabits, enter:

```
-> interfaces 2 flood rate 49
```

To change the peak flood rate for a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, **flood rate**, and the flood rate in megabits. For example, to configure the peak flood rate on port 3 on slot 2 as 49 megabits, enter:

```
-> interfaces 2/3 flood rate 49
```

To change the peak flood rate for a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **flood rate**, and the flood rate in megabits. For example, to configure the peak flood rate on ports 1 through 3 on slot 2 as 49 megabits, enter:

```
-> interfaces 2/1-3 flood rate 42
```

To specify the type of traffic eligible for rate limiting, see [“Flood Only Rate Limiting” on page 1-9](#) and [“Multicast Flood Rate Limiting” on page 1-9](#) for more information.

Configuring a Port Alias

The **interfaces alias** command is used to configure an alias (i.e., description) for a single port. (You cannot configure an entire switch or a range of ports.) To use this command, enter **interfaces** followed by the slot number, a slash (/), the port number, **alias**, and the text description, which can be up to 40 characters long.

For example, to configure an alias of “ip_phone1” for port 3 on slot 2 enter:

```
-> interfaces 2/3 alias ip_phone1
```

Note. Spaces must be contained within quotes (e.g., “IP Phone 1”).

Configuring Maximum Frame Sizes

The **interfaces max frame** command can be used to configure the maximum frame size (in bytes) on a specific port, a range of ports, or all ports on a switch. Maximum values for this command range from 1518 bytes (Ethernet packets) for Ethernet or Fast Ethernet ports to 9216 bytes (Gigabit Ethernet packets) for Gigabit Ethernet ports.

To configure the maximum frame size on an entire slot, enter **interfaces** followed by the slot number, **max frame**, and the frame size in bytes. For example, to set the maximum frame size on slot 2 to 9216 bytes, enter:

```
-> interfaces 2 max frame 9216
```

To configure the maximum frame size on a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, **max frame**, and the frame size in bytes. For example, to set the maximum frame size on port 3 on slot 2 to 9216 bytes, enter:

```
-> interfaces 2/3 max frame 9216
```

To configure the maximum frame size on a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **max frame**, and the frame size in bytes. For example, to set the maximum frame size on ports 1 through 3 on slot 2 to 9216 bytes, enter:

```
-> interfaces 2/1-3 max frame 9216
```

Setting Ethernet Parameters for Non-Combo Ports

The following sections describe how to use CLI commands to configure non-combo ports. (See the tables in [“Valid Port Settings on OmniSwitch 6250 Series Switches” on page 1-5](#) for more information.)

Setting Interface Line Speed

The **interfaces speed** command is used to set the line speed on a specific port, a range of ports, or all ports on an entire switch (slot) to one of the following parameter values:

- **10** (10 Mbps Ethernet)
- **100** (100 Mbps Fast Ethernet)
- **auto** (auto-sensing, which is the default)—The auto setting automatically detects and matches the line speed of the attached device.

Note that available settings for the **interfaces speed** command depend on the available line speeds of your hardware interface. See [“Valid Port Settings on OmniSwitch 6250 Series Switches” on page 1-5](#) for more information.

In order to set up a speed and duplex on a port, autonegotiation should be disabled.

```
-> interfaces 2 autoneg disable
```

To set the line speed on an entire switch, enter **interfaces** followed by the slot number and the desired speed. For example, to set slot 2 to 100 Mbps, enter:

```
-> interfaces 2 speed 100
```

To set the line speed on a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, and the desired speed. For example, to set the line speed on slot 2 port 3 at 100 Mbps, enter:

```
-> interfaces 2/3 speed 100
```

To set the line speed on a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, and the desired speed. For example, to set the line speed on ports 1 through 3 on slot 2 at 100 Mbps, enter:

```
-> interfaces 2/1-3 speed 100
```

Configuring Duplex Mode

The **interfaces duplex** command is used to configure the duplex mode on a specific port, a range of ports, or all ports on a switch (slot) to **full** (full duplex mode, which is the default on fiber ports), **half** (half duplex mode), and **auto** (autonegotiation, which is the default on copper ports). (The **Auto** option causes the switch to advertise all available duplex modes (half/full/both) for the port during autonegotiation.) In full duplex mode, the interface transmits and receives data simultaneously. In half duplex mode, the interface can only transmit or receive data at a given time.

Note. The **Auto** option sets both the duplex mode and line speed settings to autonegotiation.

In order to set up a speed and duplex on a port autonegotiation should be disabled.

```
-> interfaces 2 autoneg disable
```

To configure the duplex mode on an entire slot, enter **interfaces** followed by the slot number, **duplex**, and the desired duplex setting (**auto**, **full**, or **half**). For example, to set the duplex mode on slot 2 to full, enter:

```
-> interfaces 2 duplex full
```

To configure the duplex mode on a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, **duplex**, and the desired duplex setting (**auto**, **full**, or **half**). For example, to set the duplex mode on port 3 on slot 2 to full, enter:

```
-> interfaces 2/3 duplex full
```

To configure the duplex mode on a range of ports, enter **interfaces** followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **duplex**, and the desired duplex setting (**auto**, **full**, or **half**). For example, to set the duplex mode on ports 1 through 3 on slot 2 to full, enter:

```
-> interfaces 2/1-3 duplex full
```

Configuring Inter-frame Gap Values

Inter-frame gap is a measure of the minimum idle time between the end of one frame transmission and the beginning of another. By default, the inter-frame gap is 12 bytes. The **interfaces ifg** command can be used to configure the inter-frame gap value (in bytes) on a specific port, a range of ports, or all ports on a switch (slot). Values for this command range from 9 to 12 bytes.

Note. This command is only valid on Gigabit ports.

To configure the inter-frame gap on an entire slot, enter **interfaces**, followed by the slot number, **ifg**, and the desired inter-frame gap value. For example, to set the inter-frame gap value on slot 2 to 10 bytes, enter:

```
-> interfaces 2 ifg 10
```

To configure the inter-frame gap on a single port, enter **interfaces**, followed by the slot number, a slash (/), the port number, **ifg**, and the desired inter-frame gap value. For example, to set the inter-frame gap value on port 20 on slot 2 to 10 bytes, enter:

```
-> interfaces 2/20 ifg 10
```

To configure the inter-frame gap on a range of ports, enter **interfaces**, followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **ifg**, and the desired inter-frame gap value. For example, to set the inter-frame gap value on ports 20 through 22 on slot 2 to 10 bytes, enter:

```
-> interfaces 2/20-22 ifg 10
```

Configuring Autonegotiation and Crossover Settings

The following subsections describe how to enable and disable autonegotiation (see [“Enabling and Disabling Autonegotiation” on page 1-14](#)) and configure crossover settings (see [“Configuring Crossover Settings” on page 1-15](#)).

Enabling and Disabling Autonegotiation

By default, autonegotiation is enabled. To enable or disable autonegotiation on a single port, a range of ports, or an entire slot, use the **interfaces autoneg** command. (See [“Configuring Crossover Settings” on page 1-15](#) and [“Setting Ethernet Combo Port Parameters” on page 1-17](#) for more information).

To enable or disable autonegotiation on an entire switch, enter **interfaces**, followed by the slot number, **autoneg**, and either **enable** or **disable**. For example, to enable autonegotiation on slot 2, enter:

```
-> interfaces 2 autoneg enable
```

To enable or disable autonegotiation on a single port, enter **interfaces**, followed by the slot number, a slash (/), the port number, **autoneg**, and either **enable** or **disable**. For example, to enable autonegotiation on port 3 on slot 2, enter:

```
-> interfaces 2/3 autoneg enable
```

To enable or disable autonegotiation on a range of ports, enter **interfaces**, followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **autoneg**, and either **enable** or **disable**. For example, to enable autonegotiation on ports 1 through 3 on slot 2, enter:

```
-> interfaces 2/1-3 autoneg enable
```

Note. Please refer to [“Autonegotiation Guidelines” on page 1-5](#) for guidelines on configuring autonegotiation.

Configuring Crossover Settings

To configure crossover settings on a single port, a range of ports, or an entire slot, use the **interfaces crossover** command. If autonegotiation is disabled, auto MDIX, auto speed, and auto duplex are not accepted.

Setting the crossover configuration to **auto** will configure the interface or interfaces to automatically detect crossover settings. Setting crossover configuration to **mdix** will configure the interface or interfaces for MDIX (Media Dependent Interface with Crossover), which is the standard for hubs and switches. Setting crossover to **mdi** will configure the interface or interfaces for MDI (Media Dependent Interface), which is the standard for end stations.

To configure crossover settings on an entire switch, enter **interfaces**, followed by the slot number, **crossover**, and the desired setting. For example, to set the crossover configuration to auto on slot 2, enter:

```
-> interfaces 2 crossover auto
```

To configure crossover settings on a single port, enter **interfaces**, followed by the slot number, a slash (/), the port number, **crossover**, and the desired setting. For example, to set the crossover configuration to auto on port 3 on slot 2, enter:

```
-> interfaces 2/3 crossover auto
```

To configure crossover settings on a range of ports, enter **interfaces**, followed by the slot number, a slash (/), the first port number, a hyphen (-), the last port number, **crossover**, and the desired setting. For example, to set the crossover configuration to auto on ports 1 through 3 on slot 2, enter:

```
-> interfaces 2/1-3 crossover auto
```

Configuring Flow Control on Non-Combo Ports

The **interfaces pause** command is used to configure flow control (pause) settings for non-combo ports that run in full duplex mode. Configuring flow control is done to specify whether or not an interface will honor or transmit and honor PAUSE frames. PAUSE frames are used to temporarily pause the flow of traffic between two connected devices to help prevent packet loss when traffic congestion occurs between switches.

Note that if autonegotiation and flow control are both enabled for an interface, then autonegotiation determines how the interface will process PAUSE frames. See [“Flow Control and Autonegotiation” on page 1-6](#) for more information. If autonegotiation is disabled but flow control is enabled, then the configured flow control settings apply.

By default, flow control is disabled. To configure flow control for one or more ports, use the **interfaces pause** command with one of the following parameters to specify how PAUSE frames are processed:

- **rx**—Allow the interface to honor PAUSE frames from peer switches and temporarily stop sending traffic to the peer. Do not transmit PAUSE frames to peer switches.
- **tx-and-rx**—Transmit and honor PAUSE frames when traffic congestion occurs between peer switches.

For example, the following command configures ports 1/1 through 1/10 to transmit and honor PAUSE frames:

```
-> interfaces 1/1-10 pause tx-and-rx
```

To disable flow control for one or more ports, specify the **disable** parameter with the **interfaces pause** command. For example:

```
-> interfaces 1/10 pause disable
```

For more information about the **interfaces pause** command syntax, see the “Ethernet Port Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Setting Ethernet Combo Port Parameters

The following sections describe how to use CLI commands to configure combo ports on OmniSwitch 6250 switches.

Note. The commands used in this section are examples, please refer to [page 1-5](#) for the combo port numbering.

Setting Interface Line Speed for Combo Ports

The **interfaces hybrid speed** command is used to set the line speed on a specific combo port, a range of combo ports, or all combo ports on an entire switch (slot) to one of the following parameter values:

- **10** (10 Mbps Ethernet)
- **100** (100 Mbps Fast Ethernet)
- **1000** (1000 Mbps Gigabit Ethernet, which is the default for combo SFP connectors)
- **auto** (auto-sensing, which is the default for combo 10/100/1000 ports)—The **auto** setting automatically detects and matches the line speed of the attached device.

Available settings for the **interfaces hybrid speed** command depend on the available line speeds of your hardware interface. See [“Valid Port Settings on OmniSwitch 6250 Series Switches” on page 1-5](#) for more information.

Note. In the **interfaces hybrid speed** command, the **copper** keyword is used to configure the copper RJ-45 10/100/1000 port while the **fiber** keyword is used to configure the fiber SFP connectors.

To set the line speed for all combo ports on an entire switch, enter **interfaces**, followed by the slot number, **hybrid**, either **fiber** or **copper**, and the desired speed. For example, to set all combo copper ports on slot 2 to 100 Mbps, enter:

```
-> interfaces 2 hybrid copper speed 100
```

Note. using the **interfaces hybrid speed** command to set all combo ports on a switch, will not affect the configurations of the non-combo ports.

To set the line speed on a single combo port, enter **interfaces**, followed by the slot number, a slash (/), the combo port number, **hybrid**, either **fiber** or **copper**, and the desired speed. For example, to set the line speed on slot 2 combo copper RJ-45 port 25 to 100 Mbps, enter:

```
-> interfaces 2/25 hybrid copper speed 100
```

To set the line speed on a range of combo ports, enter **interfaces**, followed by the slot number, a slash (/), the first combo port number, a hyphen (-), the last combo port number, **hybrid**, either **fiber** or **copper**, and the desired speed. For example, to set the line speed on combo copper ports 25 through 26 on slot 2 to 100 Mbps, enter:

```
-> interfaces 2/25-26 hybrid copper speed 100
```

Configuring Duplex Mode for Combo Ports

The **interfaces hybrid duplex** command is used to configure the duplex mode on a specific combo port, a range of combo ports, or all combo ports on a switch (slot) to **full** (full duplex mode, which is the default for 100 Mbps fiber SFP and 1 Gbps fiber SFP), **half** (half duplex mode), **auto** (auto-negotiation, which is the default for copper RJ-45 ports). (The **Auto** option sets both the duplex mode and line speed settings to autonegotiation.) In full duplex mode, the interface transmits and receives data simultaneously. In half duplex mode, the interface can only transmit or receive data at a given time. (Available settings for this command depend on the available line speeds of your hardware interface. See [“Valid Port Settings on OmniSwitch 6250 Series Switches” on page 1-5](#) for more information.)

Note. In the **interfaces hybrid duplex** command the **copper** keyword is used to configure the copper RJ-45 10/100/1000 port while the **fiber** keyword is used to configure the fiber SFP connector.

To configure the duplex mode on an entire slot, enter **interfaces**, followed by the slot number, **hybrid**, either **fiber** or **copper**, **duplex**, and the desired duplex setting (**auto**, **full**, or **half**). For example, to set the duplex mode on all fiber combo ports on slot 2 to full, enter:

```
-> interfaces 2 hybrid fiber duplex full
```

Note. using the **interfaces hybrid duplex** command to set all combo ports on a switch, will not affect the configurations of the non-combo ports.

To configure the duplex mode on a single combo port, enter **interfaces**, followed by the slot number, a slash (/), the combo port number, **hybrid**, either **fiber** or **copper**, **duplex**, and the desired duplex setting (**auto**, **full**, or **half**). For example, to set the duplex mode on the fiber combo port 23 on slot 2 to full, enter:

```
-> interfaces 2/25 hybrid fiber duplex full
```

To configure the duplex mode on a range of combo ports, enter **interfaces**, followed by the slot number, a slash (/), the first combo port number, a hyphen (-), the last combo port number, **hybrid**, either **fiber** or **copper**, **duplex**, and the desired duplex setting (**auto**, **full**, or **half**). For example, to set the duplex mode on fiber combo ports 25 through 26 on slot 2 to full, enter:

```
-> interfaces 2/25-26 hybrid fiber duplex full
```

Configuring Autonegotiation and Crossover for Combo Ports

The following subsections describe how to enable and disable autonegotiation (see [“Enabling and Disabling Autonegotiation for Combo Ports” on page 1-18](#)) and configure crossover settings (see [“Configuring Crossover Settings for Combo Ports” on page 1-19](#)) on combo ports.

Enabling and Disabling Autonegotiation for Combo Ports

By default, autonegotiation is enabled. To enable or disable autonegotiation on a single combo port, a range of combo ports, or all combo ports on an entire switch (slot), use the **interfaces hybrid autoneg** command. (See [“Configuring Crossover Settings for Combo Ports” on page 1-19](#) for more information).

Note. In the **interfaces hybrid autoneg** command, the **copper** keyword is used to configure the copper RJ-45 10/100/1000 port while the **fiber** keyword is used to configure the fiber SFP connector.

To enable or disable autonegotiation on all combo ports in an entire switch, enter **interfaces**, followed by the slot number, **hybrid**, either **fiber** or **copper**, **autoneg**, and either **enable** or **disable**. For example, to enable autonegotiation on all copper combo ports on slot 2, enter:

```
-> interfaces 2 hybrid copper autoneg enable
```

Note. using the **interface hybrid autoneg** command to set all combo ports on a switch will not affect the configurations of the non-combo ports.

To enable or disable autonegotiation on a single combo port, enter **interfaces**, followed by the slot number, a slash (/), the combo port number, **hybrid**, either **fiber** or **copper**, **autoneg**, and either **enable** or **disable**. For example, to enable autonegotiation on copper combo port 25 on slot 2, enter:

```
-> interfaces 2/25 hybrid copper autoneg enable
```

To enable or disable autonegotiation on a range of combo ports, enter **interfaces**, followed by the slot number, a slash (/), the first combo port number, a hyphen (-), the last combo port number, **hybrid**, either **fiber** or **copper**, **autoneg**, and either **enable** or **disable**. For example, to enable autonegotiation on copper combo ports 25 through 26 on slot 2, enter:

```
-> interfaces 2/25-26 hybrid copper autoneg enable
```

As an option, you can document the interface type by entering **ethernet**, **fastethernet**, or **gigaethernet** before the slot number. For example, to enable autonegotiation on copper combo port 23 on slot 2 and document the combo port as Gigabit Ethernet, enter:

```
-> interfaces gigaethernet 2/23 hybrid copper autoneg enable
```

Note. Please refer to [“Autonegotiation Guidelines” on page 1-5](#) for guidelines on configuring autonegotiation.

Configuring Crossover Settings for Combo Ports

To configure crossover settings on a single combo port, a range of combo ports, or all combo ports in an entire switch (slot), use the **interfaces hybrid crossover** command. If autonegotiation is disabled, auto MDIX, auto speed, and auto duplex are not accepted.

Note. In the **interfaces hybrid crossover** command, the **copper** keyword is used to configure the copper RJ-45 10/100/1000 port.

Setting the crossover configuration to **auto** will configure the interface or interfaces to automatically detect crossover settings. Setting crossover configuration to **mdix** will configure the interface or interfaces for MDIX (Media Dependent Interface with Crossover), which is the standard for hubs and switches. Setting crossover to **mdi** will configure the interface or interfaces for MDI (Media Dependent Interface), which is the standard for end stations.

To configure crossover settings for all combo ports on an entire switch, enter **interfaces**, followed by the slot number, **hybrid**, **copper**, **crossover**, and the desired setting. For example, to set the crossover configuration to auto on for all copper combo ports slot 2, enter:

```
-> interfaces 2 hybrid copper crossover auto
```

Note. using the **interface hybrid crossover** command to set all combo ports on a switch will not affect the configurations of the non-combo ports.

To configure crossover settings on a single combo port, enter **interfaces**, followed by the slot number, a slash (/), the combo port number, **hybrid**, **copper**, **crossover**, and the desired setting. For example, to set the crossover configuration to auto on copper combo port 23 on slot 2, enter:

```
-> interfaces 2/25 hybrid copper crossover auto
```

To configure crossover settings on a range of combo ports, enter **interfaces**, followed by the slot number, a slash (/), the first combo port number, a hyphen (-), the last combo port number, **hybrid**, **copper**, **crossover**, and the desired setting. For example, to set the crossover configuration to auto on copper combo ports 25 through 26 on slot 2, enter:

```
-> interfaces 2/25-26 hybrid copper crossover auto
```

Configuring Flow Control on Combo Ports

The **interfaces hybrid pause** command is used to configure flow control (pause) settings for combo ports that run in full duplex mode. Configuring flow control is done to specify whether or not an interface will honor or both transmit and honor PAUSE frames. PAUSE frames are used to temporarily pause the flow of traffic between two connected devices to help prevent packet loss when traffic congestion occurs between switches.

Note that if autonegotiation and flow control are both enabled for an interface, then autonegotiation determines how the interface will process PAUSE frames. See [“Flow Control and Autonegotiation” on page 1-6](#) for more information. If autonegotiation is disabled but flow control is enabled, then the configured flow control settings apply.

By default, flow control is disabled. To configure flow control for one or more ports, use the **interfaces hybrid pause** command with one of the following parameters to specify how PAUSE frames are processed:

- **rx**—Allow the interface to honor PAUSE frames from peer switches and temporarily stop sending traffic to the peer. Do not transmit PAUSE frames to peer switches.
- **tx-and-rx**—Transmit and honor PAUSE frames when traffic congestion occurs between peer switches.

Note. In the **interfaces hybrid pause** command, the **copper** keyword is used to configure the copper RJ-45 10/100/1000 port while the **fiber** keyword is used to configure the fiber SFP connector.

For example, the following command configures port 1/25 to transmit and honor PAUSE frames:

```
-> interfaces 1/25 hybrid fiber pause tx-and-rx
```

To disable flow control, use the **disable** parameter with the **interfaces hybrid pause** command.
For example:

```
-> interfaces 1/25 hybrid fiber pause disable
```

For more information about the **interfaces hybrid pause** command syntax, see the “Ethernet Port Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Verifying Ethernet Port Configuration

To display information about Ethernet port configuration settings, use the following **show** commands:

show interfaces pause	Displays the flow control pause configuration for switch interfaces.
show interfaces	Displays general interface information, such as hardware, MAC address, input and output errors.
show interfaces accounting	Displays interface accounting information.
show interfaces counters	Displays interface counters information.
show interfaces counters errors	Displays interface error frame information for Ethernet and Fast Ethernet ports.
show interfaces collisions	Displays collision statistics information for Ethernet and Fast Ethernet ports.
show interfaces status	Displays line status information.
show interfaces port	Displays port status information.
show interfaces ifg	Displays inter-frame gap values.
show interfaces flood rate	Displays peak flood rate settings.
show interfaces traffic	Displays interface traffic statistics.
show interfaces capability	Displays autonegotiation, flow, speed, duplex, and crossover settings.
show interfaces hybrid	Displays general interface information (e.g., hardware, MAC address, input errors, output errors) for combo ports.
show interfaces hybrid status	Displays line status information for combo ports.
show interfaces hybrid flow control	Displays interface flow control wait time settings in nanoseconds for combo ports.
show interfaces hybrid pause	Displays the flow control pause configuration for combo ports.
show interfaces hybrid capability	Displays autonegotiation, flow, speed, duplex, and crossover settings for combo ports.
show interfaces hybrid accounting	Displays interface accounting information (e.g., packets received/transmitted, deferred frames received) for combo ports.
show interfaces hybrid counters	Displays interface counters information (e.g., unicast, broadcast, multi-cast packets received/transmitted) for combo ports.
show interfaces hybrid counters errors	Displays interface error frame information (e.g., CRC errors, transit errors, receive errors) for combo ports.
show interfaces hybrid collisions	Displays interface collision information (e.g., number of collisions, number of retries) for combo ports.
show interfaces hybrid traffic	Displays interface traffic statistics for combo ports.
show interfaces hybrid port	Displays interface port status (up or down) for combo ports.
show interfaces hybrid flood rate	Displays interface peak flood rate settings for combo ports.
show interfaces hybrid ifg	Displays interface inter-frame gap values for combo ports.

These commands can be quite useful in troubleshooting and resolving potential configuration issues or problems on your switch. For more information about the resulting displays from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

2 Managing Source Learning

Transparent bridging relies on a process referred to as *source learning* to handle traffic flow. Network devices communicate by sending and receiving data packets that each contain a source MAC address and a destination MAC address. When packets are received on switch network interface (NI) module ports, source learning examines each packet and compares the source MAC address to entries in a MAC address database table. If the table does not contain an entry for the source address, then a new record is created associating the address with the port it was learned on. If an entry for the source address already exists in the table, a new one is not created.

Packets are also filtered to determine if the source and destination address are on the same LAN segment. If the destination address is not found in the MAC address table, then the packet is forwarded to all other switches that are connected to the same LAN. If the MAC address table does contain a matching entry for the destination address, then there is no need to forward the packet to the rest of the network.

In This Chapter

This chapter describes how to manage source learning entries in the switch MAC address table (often referred to as the *forwarding or filtering database*) through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

Configuration procedures described in this chapter include:

- [“Using Static MAC Addresses” on page 2-5.](#)
- [“Using Static Multicast MAC Addresses” on page 2-7](#)
- [“Configuring MAC Address Table Aging Time” on page 2-9.](#)
- [“Configuring the Source Learning Status” on page 2-10.](#)
- [“Displaying Source Learning Information” on page 2-11.](#)

Source Learning Specifications

The functionality described in this chapter is supported on the OmniSwitch 6250 Series unless otherwise stated in the following Specifications table or specifically noted within any section of this chapter.

RFCs supported	2674— <i>Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions</i>
IEEE Standards supported	802.1Q— <i>Virtual Bridged Local Area Networks</i> 802.1D— <i>Media Access Control Bridges</i>
Maximum number of learned MAC addresses when synchronized MAC source learning mode is enabled	OmniSwitch 6250 = 16K/stack
Maximum number of static L2 multicast MAC addresses.	OmniSwitch 6250 = 256/stack

Source Learning Defaults

Parameter Description	Command	Default
Static MAC address management status	mac-address-table	permanent
Static MAC address operating mode	mac-address-table	bridging
MAC address aging timer	mac-address-table aging-time	300 seconds
MAC source learning status per port	source-learning	

Sample MAC Address Table Configuration

The following steps provide a quick tutorial that will create a static MAC address and change the MAC address aging timer for VLAN 200:

Note. Optional. Creating a static MAC address involves specifying an address that is not already used in another static entry or already dynamically learned by the switch. To determine if the address is already known to the MAC address table, enter **show mac-address-table**. If the address does not appear in the **show mac-address-table** output, then it is available to use for configuring a static MAC address entry. For example,

```
-> show mac-address-table
Legend: Mac Address: * = address not valid
```

Vlan	Mac Address	Type	Protocol	Operation	Interface
1	00:00:00:00:00:01	learned	0800	bridging	8/ 1
1	00:d0:95:6a:73:9a	learned	aaaa0003	bridging	10/23

Total number of Valid MAC addresses above = 2

The **show mac-address-table** command is also useful for monitoring general source learning activity and verifying dynamic VLAN assignments of addresses received on mobile ports.

1 Create VLAN 200, if it does not already exist, using the following command:

```
-> vlan 200
```

2 Assign switch ports 2 through 5 on slot 3 to VLAN 200—if they are not already associated with VLAN 200—using the following command:

```
-> vlan 200 port default 3/2-5
```

3 Create a static MAC address entry using the following command to assign address 002D95:5BF30E to port 3/4 associated with VLAN 200 and to specify a permanent management status for the static address:

```
-> mac-address-table permanent 00:2d:95:5b:f3:0e 3/4 200
```

4 Change the MAC address aging time to 500 seconds (the default is 300 seconds) using the following command:

```
-> mac-address-table aging-time 500
```

Note. Optional. To verify the static MAC address configuration, enter **show mac-address-table**. For example:

```
-> show mac-address-table
Legend: Mac Address: * = address not valid
```

Vlan	Mac Address	Type	Protocol	Operation	Interface
1	00:00:00:00:00:01	learned	0800	bridging	8/1
1	00:d0:95:6a:73:9a	learned	aaaa0003	bridging	10/23
200	00:2d:95:5b:f3:0e	delontimeout	0	bridging	3/4

Total number of Valid MAC addresses above = 3

To verify the new aging time value, enter **show mac-address-table aging-time**. For example,

```
-> show mac-address-table aging-time  
Mac Address Aging Time (seconds) = 300
```

MAC Address Table Overview

Source learning builds and maintains the MAC address table on each switch. New MAC address table entries are created in one of two ways: they are dynamically learned or statically assigned. Dynamically learned MAC addresses are those that are obtained by the switch when source learning examines data packets and records the source address and the port and VLAN it was learned on. Static MAC addresses are user defined addresses that are statically assigned to a port and VLAN using the [mac-address-table](#) command.

Accessing MAC Address Table entries is useful for managing traffic flow and troubleshooting network device connectivity problems. For example, if a workstation connected to the switch is unable to communicate with another workstation connected to the same switch, the MAC address table might show that one of these devices was learned on a port that belonged to a different VLAN or the source MAC address of one of the devices may not appear at all in the address table.

Using Static MAC Addresses

Static MAC addresses are configured using the [mac-address-table](#) command. These addresses direct network traffic to a specific port and VLAN. They are particularly useful when dealing with silent network devices. These types of devices do not send packets, so their source MAC address is never learned and recorded in the MAC address table. Assigning a MAC address to the silent device's port creates a record in the MAC address table and ensures that packets destined for the silent device are forwarded out that port.

When defining a static MAC address for a particular slot/port and VLAN, consider the following:

- Configuring static MAC addresses is only supported on non-mobile ports.
- The specified slot/port must already belong to the specified VLAN. Use the [vlan port default](#) command to assign a port to a VLAN before you configure the static MAC address.
- Only traffic from other ports associated with the same VLAN is directed to the static MAC address slot/port.
- Static MAC addresses are **permanent** addresses. This means that a static MAC address remains in use even if the MAC ages out or the switch is rebooted.
- There are two types of static MAC address behavior supported: **bridging** (default) or **filtering**. Enter **filtering** to set up a denial of service to block potential hostile attacks. Traffic sent to or from a filtered MAC address is dropped. Enter **bridging** for regular traffic flow to or from the MAC address. For more information about Layer 2 filtering, see [Chapter 34, "Configuring QoS."](#)
- If a packet received on a port associated with the same VLAN contains a source address that matches a static MAC address, the packet is discarded. The same source address on different ports within the same VLAN is not supported.
- If a static MAC address is configured on a port link that is down or disabled, an asterisk appears to the right of the MAC address in the [show mac-address-table](#) command display. The asterisk indicates that this is an invalid MAC address. When the port link comes up, however, the MAC address is then considered valid and the asterisk no longer appears next to the address in the display.

Configuring Static MAC Addresses

To configure a permanent, bridging static MAC address, enter **mac-address-table** followed by a MAC address, slot/port, and the VLAN ID to assign to the MAC address. For example, the following assigns a MAC address to port 10 on slot 4 associated with VLAN 255:

```
-> mac-address-table 00:02:DA:00:59:0C 4/10 255
```

Since **permanent** and **bridging** options for a static MAC are default settings, it is not necessary to enter them as part of the command.

Use the **no** form of this command to clear MAC address entries from the table. If the MAC address status type (permanent or learned) is not specified, then only permanent addresses are removed from the table. The following example removes a MAC address entry that is assigned on port 2 of slot 3 for VLAN 855 from the MAC address table:

```
-> no mac-address-table 00:00:02:CE:10:37 3/2 855
```

If a slot/port and VLAN ID are not specified when removing MAC address table entries, then all MACs defined with the specified status are removed. For example, the following command removes all learned MAC addresses from the table, regardless of their slot/port or VLAN assignments:

```
-> no mac-address-table learned
```

To verify static MAC address configuration and other table entries, use the **show mac-address-table** command. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Static MAC Addresses on Link Aggregate Ports

Static MAC Addresses are not assigned to physical ports that belong to a link aggregate. Instead, they are assigned to a link aggregate ID that represents a collection of physical ports. This ID is specified at the time the link aggregate of ports is created and when using the **mac-address-table** command.

To configure a permanent, bridging static MAC address on a link aggregate ID, enter **mac-address-table** followed by a MAC address, then **linkagg** followed by the link aggregate ID, and the VLAN ID to assign to the MAC address. For example, the following assigns a MAC address to link aggregate ID 2 associated with VLAN 455:

```
-> mac-address-table 00:95:2A:00:3E:4C linkagg 2 455
```

For more information about configuring a link aggregate of ports, see [Chapter 23, “Configuring Static Link Aggregation”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Using Static Multicast MAC Addresses

Using static multicast MAC addresses allows you to send traffic intended for a single destination multicast MAC address to selected switch ports within a given VLAN. To specify which ports will receive the multicast traffic, a static multicast address is assigned to each selected port for a given VLAN. The ports associated with the multicast address are then identified as egress ports. When traffic received on ports within the same VLAN is destined for the multicast address, the traffic is forwarded only on the egress ports that are associated with the multicast address.

When defining a static multicast MAC address for a particular port and VLAN, consider the following:

- A MAC address is considered a multicast MAC address if the least significant bit of the most significant octet of the address is enabled. For example, MAC addresses with a prefix of 01, 03, 05, 13, etc., are multicast MAC addresses.
- If a multicast prefix value is not present, then the address is treated as a regular MAC address and not allowed when using the **mac-address-table static-multicast** command.
- Multicast addresses within the following ranges are not supported:
01:00:5E:00:00:00 to 01:00:5E:7F:FF:FF
01:80:C2:XX.XX.XX
33:33:XX:XX:XX:XX
- Configuring static multicast addresses is only supported on non-mobile ports.
- In addition to configuring the same static multicast address for multiple ports within a given VLAN, it is also possible to use the same multicast address across multiple VLANs.
- The specified port or link aggregate ID must already belong to the specified VLAN. Use the **vlan port default** command to assign a port or link aggregate to a VLAN before you configure the static multicast address.

Configuring Static Multicast MAC Addresses

The **mac-address-table static-multicast** command is used to define a destination multicast MAC address and assign the address to one or more egress ports within a specified VLAN. For example, the following command assigns the multicast address 01:25:9a:5c:2f:10 to port 1/24 in VLAN 20:

```
-> mac-address-table static-multicast 01:25:9a:5c:2f:10 1/24 20
```

To assign a multicast address to more than one port, enter a range of ports and/or multiple port entries on the same command line separated by a space. For example, the following command assigns the multicast address 01:25:9a:5c:2f:10 to port 1/24 and ports 2/1 through 2/6 in VLAN 20:

```
-> mac-address-table static-multicast 01:25:9a:5c:2f:10 1/24 2/1-6 20
```

Use the **no** form of the **mac-address-table static-multicast** command to delete static multicast MAC address entries. For example, the following command deletes a static multicast address that is assigned to port 2 on slot 3 for VLAN 855:

```
-> no mac-address-table static-multicast 01:00:02:CE:10:37 3/2 855
```

If a MAC address, slot/port and VLAN ID are not specified with this form of the command, then all static multicast addresses are deleted. For example, the following command deletes all static MAC addresses, regardless of their slot/port or VLAN assignments:

```
-> no mac-address-table static-multicast
```

To verify the static MAC address configuration and other table entries, use the **show mac-address-table** and **show mac-address-table static-multicast** commands. For more information about these commands, see the *OmniSwitch CLI Reference Guide*.

Static Multicast MAC Addresses on Link Aggregate Ports

Static multicast MAC addresses are not assigned to physical ports that belong to a link aggregate. Instead, they are assigned to a link aggregate ID that represents a collection of physical ports. This ID is specified at the time the link aggregate of ports is created and when using the **mac-address-table static-multicast** command.

To configure a static multicast MAC address on a link aggregate ID, use the **mac-address-table static-multicast** command with the **linkagg** keyword to specify the link aggregate ID. For example, the following command assigns a static multicast MAC address to link aggregate ID 2 associated with VLAN 455:

```
-> mac-address-table static-multicast 01:95:2A:00:3E:4C linkagg 2 455
```

For more information about configuring a link aggregate of ports, see Chapter 13, “Configuring Static Link Aggregation” and Chapter 14, “Configuring Dynamic Link Aggregation.”

ASCII-File-Only Syntax

When a static multicast MAC address is configured and saved (typically through the **snapshot** or **write memory** commands), the **mac-address-table static-multicast** command captured in the ASCII text file or **boot.cfg** file will include an additional **group** parameter. This parameter indicates the number of the multicast group that the switch has assigned to the multicast MAC address for the given VLAN association. For example:

```
-> mac-address-table static-multicast 01:25:9a:5c:2f:10 1/24 2/1-6 20 group 1
```

In this example, the multicast MAC address, 01:25:9a:5c:2f:10, is associated with ports 1/24 and 2/1 through 2/6 in VLAN 20. The additional **group** parameter value shown in the example indicates that the switch will assign the multicast-VLAN association created with the **mac-address-table static-multicast** to multicast group one.

Note that if the port assigned to a multicast MAC address is down or administratively disabled when the **configuration snapshot** or **write memory** command is used, the multicast MAC address is not saved to the resulting ASCII file or **boot.cfg** file.

Each multicast MAC address association with a VLAN is treated as a unique instance and is assigned a multicast group number specific to that instance. This is also the case when the same multicast address is associated with more than one VLAN; each VLAN association is assigned a multicast group number even though the MAC address is the same for each instance. Note that up to 1022 multicast address-VLAN associations are supported per switch.

Configuring MAC Address Table Aging Time

Source learning also tracks MAC address age and removes addresses from the MAC address table that have aged beyond the aging timer value. When a device stops sending packets, source learning keeps track of how much time has passed since the last packet was received on the device's switch port. When this amount of time exceeds the aging time value, the MAC is *aged out* of the MAC address table. Source learning always starts tracking MAC address age from the time since the last packet was received.

By default, the aging time is set to 300 seconds (5 minutes) and is configured on a global basis using the **mac-address-table aging-time** command. For example, the following sets the aging time for all VLANs to 500 seconds:

```
-> mac-address-table aging-time 500
```

A MAC address learned on any VLAN port will age out if the time since a packet with that address was last seen on the port exceeds 500 seconds.

Note. An inactive MAC address may take up to twice as long as the aging time value specified to age out of the MAC address table. For example, if an aging time of 60 seconds is specified, the MAC will age out any time between 60 and 120 seconds of inactivity.

When using the **mac-address-table aging-time** command in a switch configuration file (e.g., **boot.cfg**), include an instance of this command specifying the VLAN ID for each VLAN configured on the switch. This is necessary even though all VLANs will have the same aging time value.

To set the aging time back to the default value, use the **no** form of the **mac-address-table aging-time** command. For example, the following sets the aging time for all VLANs back to the default of 300 seconds:

```
-> no mac-address-table aging-time
```

Note. The MAC address table aging time is also used as the timeout value for the Address Resolution Protocol (ARP) table. This timeout value determines how long the switch retains dynamically learned ARP table entries. See [Chapter 25, "Configuring IP,"](#) for more information.

To display the aging time value for one or all VLANs, use the **show mac-address-table aging-time** command. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Configuring the Source Learning Status

The source learning status for a port or link aggregate of ports is configurable using the **source-learning** command. By default, source learning is enabled on a port or link aggregate. To disable the status, use the **source-learning** command with the **disable** option. For example:

```
-> source-learning port 1/10 disable
-> source-learning port 1/15-20 disable
-> source-learning linkagg 10 disable
```

To enable the source learning status for a port or link aggregate, use the **source-learning** command with the **enable** option. For example:

```
-> source-learning port 1/10 enable
-> source-learning port 1/15-20 enable
-> source-learning linkagg 10 enable
```

Disabling source learning on a port or link aggregate is useful on a ring configuration, where a switch within the ring does not need to learn the MAC addresses that the same switch is forwarding to another switch within the ring. This functionality is also useful in Transparent LAN Service configurations, where the service provider device does not need to learn the MAC addresses of the customer network.

Configuring the source learning status is not allowed on the following types of switch ports:

- Mobile ports, including 802.1X ports (802.1X is enabled on mobile ports only).
- Ports enabled with Learned Port Security (LPS).
- Member ports of a link aggregate.

Consider the following guidelines when changing the source learning status for a port or link aggregate:

- Disabling source learning on a link aggregate disables MAC address learning on all member ports of the link aggregate.
- MAC addresses dynamically learned on a port or aggregate are cleared when source learning is disabled.
- Statically configured MAC addresses are not cleared when source learning is disabled for the port or aggregate. In addition, configuring a new static MAC address is allowed even when source learning is disabled.

Displaying Source Learning Information

To display MAC Address Table entries, statistics, and aging time values, use the show commands listed below:

show mac-address-table	Displays a list of all MAC addresses known to the MAC address table, including static MAC addresses.
show mac-address-table static-multicast	Displays a list of all static multicast MAC addresses known to the MAC address table. Note that only static multicast addresses assigned to ports that are up and enabled are displayed with this command.
show mac-address-table count	Displays a count of the different types of MAC addresses (learned, permanent, reset, and timeout). Also includes a total count of all addresses known to the MAC address table.
show mac-address-table aging-time	Displays the current MAC address aging timer value by switch or VLAN.

For more information about the resulting displays from these commands, see the *OmniSwitch CLI Reference Guide*. An example of the output for the **show mac-address-table** and **show mac-address-table aging-time** commands is also given in “[Sample MAC Address Table Configuration](#)” on page 2-3.

3 Configuring Learned Port Security

Learned Port Security (LPS) provides a mechanism for authorizing source learning of MAC addresses on Ethernet and Gigabit Ethernet ports. The only types of Ethernet ports that LPS does not support are link aggregate and tagged (trunked) link aggregate ports. Using LPS to control source MAC address learning provides the following benefits:

- A configurable source learning time limit that applies to all LPS ports.
- A configurable limit on the number of MAC addresses allowed on an LPS port.
- Dynamic configuration of a list of authorized source MAC addresses.
- Static configuration of a list of authorized source MAC addresses.
- Two methods for handling unauthorized traffic: stopping all traffic on the port or only blocking traffic that violates LPS criteria.

In This Chapter

This chapter describes how to configure LPS parameters through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling LPS for a port on [page 3-7](#).
- Specifying a source learning time limit for all LPS ports on [page 3-8](#).
- Configuring the maximum number of MAC addresses learned per port on [page 3-9](#).
- Configuring the maximum number of filtered MAC addresses learned per port on [page 3-10](#).
- Configuring a list of authorized MAC addresses for an LPS port on [page 3-10](#).
- Configuring a range of authorized MAC addresses for an LPS port on [page 3-10](#).
- Selecting the security violation mode for an LPS port on [page 3-11](#).
- Displaying LPS configuration information on [page 3-12](#).

For more information about source MAC address learning, see [Chapter 2, “Managing Source Learning.”](#)

Learned Port Security Specifications

RFCs supported	Not applicable at this time.
IEEE Standards supported	Not applicable at this time.
Platforms Supported	OmniSwitch 6250 Series
Ports eligible for Learned Port Security	Ethernet and gigabit Ethernet ports (fixed, mobile, 802.1Q tagged, and authenticated ports).
Ports not eligible for Learned Port Security	Link aggregate ports. 802.1Q (trunked) link aggregate ports.
Minimum number of learned MAC addresses allowed per port	1
Maximum number of learned MAC addresses allowed per port	100
Maximum number of configurable MAC address ranges per LPS port	1
Maximum number of learned MAC addresses per switch	16K

Learned Port Security Defaults

Parameter Description	Command	Default
LPS status for a port.	port-security	disabled
Number of learned MAC addresses allowed on an LPS port.	port-security maximum	1
Maximum number of filtered MAC addresses that the LPS port can learn.	port-security max-filtering	5
Source learning time limit.	port-security shutdown	disabled
Configured MAC addresses per LPS port.	port-security mac	none
MAC address range per LPS port.	port-security mac-range	00:00:00:00:00:00–ff:ff:ff:ff:ff:ff
LPS port violation mode.	port-security violation	restrict
Number of bridged MAC addresses learned before a trap is sent.	port-security learn-trap-threshold	5

Sample Learned Port Security Configuration

This section provides a quick tutorial that demonstrates the following tasks:

- Enabling LPS on a set of switch ports.
- Defining the maximum number of learned MAC addresses allowed on an LPS port.
- Defining the time limit in which source learning is allowed on all LPS ports.
- Selecting a method for handling unauthorized traffic received on an LPS port.

Note that LPS is supported on Ethernet and gigabit Ethernet fixed, mobile, tagged and authenticated ports. Link aggregate and tagged (trunked) link aggregate ports are not eligible for LPS monitoring and control.

1 Enable LPS on ports 6 through 12 on slot 3, 4, and 5 using the following command:

```
-> port-security 3/6-12 4/6-12 5/6-12 enable
```

2 Set the total number of learned MAC addresses allowed on the same ports to 25 using the following command:

```
-> port-security 3/6-12 4/6-12 5/6-12 maximum 25
```

3 Configure the amount of time in which source learning is allowed on all LPS ports to 30 minutes using the following command:

```
-> port-security shutdown 30
```

4 Select **shutdown** for the LPS violation mode using the following command:

```
-> port-security 3/6-12 4/6-12 5/6-12 violation shutdown
```

Note. *Optional.* To verify LPS port configurations, use the [port-security learn-trap-threshold](#) command. For example:

```
-> show port-security
```

```
Port: 1/30
  Operation Mode      :      DISABLED,
  Max Bridged MAC allowed :          1,
  Max Filtered MAC allowed :          5,
  Low End of MAC Range  : 00:00:00:00:00:00,
  High End of MAC Range : ff:ff:ff:ff:ff:ff,
  Violation Setting    :      RESTRICT,
```

```

      MAC          VLAN      MAC TYPE
-----+-----+-----
 00:20:95:00:fa:5c    1      STATIC
```

To verify the new source learning time limit value, use the [show port-security shutdown](#) command. For example:

```
-> show port-security shutdown
LPS Shutdown Config      = 2 min
Convert-to-static        = DISABLE
Remaining Learning Window = 110 sec
```

Learned Port Security Overview

Learned Port Security (LPS) provides a mechanism for controlling network device access on one or more switch ports. Configurable LPS parameters allow the user to restrict the source learning of host MAC addresses to:

- A specific amount of time in which the switch allows source learning to occur on all LPS ports.
- A maximum number of learned MAC addresses allowed on the port.
- A list of configured authorized source MAC addresses allowed on the port.

Additional LPS functionality allows the user to specify how the LPS port handles unauthorized traffic. The following two options are available for this purpose:

- Block only traffic that violates LPS port restrictions; authorized traffic is forwarded on the port.
- Disable the LPS port when unauthorized traffic is received; all traffic is stopped and a port reset is required to return the port to normal operation.

LPS functionality is supported on the following Ethernet and Gigabit Ethernet port types:

- Fixed (non-mobile)
- Mobile
- 802.1Q tagged
- Authenticated
- 802.1x

The following port types are not supported:

- Link aggregate
- Tagged (trunked) link aggregate

How LPS Authorizes Source MAC Addresses

When a packet is received on a port that has LPS enabled, switch software checks the following criteria to determine if the source MAC address contained in the packet is allowed on the port:

- Is the source learning time window open?
- Is the number of MAC addresses learned on the port below the maximum number allowed?
- Is there a configured authorized MAC address entry for the LPS port that matches the packet's source MAC address?

Using the above criteria, the following table shows the conditions under which a MAC address is learned or blocked on an LPS port:

Time Limit	Max Number	Configured MAC	Result
Open	Below	No entry	No LPS violation; MAC learned
Closed	Below	No entry	LPS violation; MAC blocked
Open	Above	No entry	LPS violation; MAC blocked
Open	Below	Yes; entry matches	No LPS violation; MAC learned
Closed	Below	Yes; entry matches	No LPS violation; MAC learned
Open	Above	Yes; entry matches	LPS violation; MAC blocked
Open	Below	Yes; entry doesn't match	No LPS violation; MAC learned
Closed	Below	Yes; entry doesn't match	LPS violation; MAC blocked
Open	Above	Yes; entry doesn't match	LPS violation; MAC blocked

When a source MAC address violates any of the LPS conditions, the address is considered unauthorized. The LPS violation mode determines if the unauthorized MAC address is simply blocked (filtered) on the port or if the entire port is disabled (see [“Selecting the Security Violation Mode” on page 3-11](#)). Regardless of which mode is selected, notice is sent to the Switch Logging task to indicate that a violation has occurred.

Dynamic Configuration of Authorized MAC Addresses

Once LPS authorizes the learning of a source MAC address, an entry containing the address and the port it was learned on is made in an LPS database table. This entry is then used as criteria for authorizing future traffic from this source MAC on that same port. In other words, learned authorized MAC addresses become configured criteria for an LPS port.

For example, if the source MAC address 00:da:95:00:59:0c is received on port 2/10 and meets the LPS restrictions defined for that port, then this address and its port are recorded in the LPS table. All traffic that is received on port 2/10 is compared to the 00:da:95:00:59:0c entry. If any traffic received on this port consists of packets that do not contain a matching source address, the packets are then subject to the LPS source learning time limit window and the maximum number of addresses allowed criteria.

When a dynamically configured MAC address is added to the LPS table, it does not become a configured MAC address entry in the LPS table until the switch configuration file is saved and the switch is rebooted. If a reboot occurs before this is done, all dynamically learned MAC addresses in the LPS table are cleared.

Static Configuration of Authorized MAC Addresses

It is also possible to statically configure authorized source MAC address entries into the LPS table. This type of entry behaves the same way as dynamically configured entries in that it authorizes port access to traffic that contains a matching source MAC address.

Static source MAC address entries, however, take precedence over dynamically learned entries. For example, if there are 2 static MAC address entries configured for port 2/1 and the maximum number allowed on port 2/1 is 10, then only 8 dynamically learned MAC addresses are allowed on this port.

Note that source learning of configured authorized MAC addresses is still allowed after the LPS time limit has expired. However, all learning is stopped if the number of MAC addresses learned meets or exceeds the maximum number of addresses allowed, even if the LPS time limit has not expired.

There are two ways to define a static source MAC address entry in the LPS table; specify an individual MAC address or a range of MAC addresses. See [“Configuring Authorized MAC Addresses” on page 3-10](#) and [“Configuring an Authorized MAC Address Range” on page 3-10](#) for more information.

Note. Statically configured authorized MAC addresses are displayed permanently in the MAC address table for the specified LPS port; they will not be learned on any other port in the same VLAN.

Understanding the LPS Table

The LPS database table is separate from the source learning MAC address table. However, when a MAC is authorized for learning on an LPS port, an entry is made in the MAC address table in the same manner as if it was learned on a non-LPS port (see [Chapter 2, “Managing Source Learning,”](#) for more information).

In addition to dynamic and configured source MAC address entries, the LPS table also provides the following information for each eligible LPS port:

- The LPS status for the port; enabled or disabled.
- The maximum number of MAC addresses allowed on the port.
- The maximum number of MAC addresses that can be filtered on the port.
- The violation mode selected for the port; restrict or shutdown.
- Statically configured MAC addresses and MAC address ranges.
- All MAC addresses learned on the port.
- The management status for the MAC address entry; configured or dynamic.

Note that dynamic MAC address entries become configured entries after the switch configuration is saved and the switch is rebooted. However, any dynamic MAC address entries that are not saved to the switch configuration are cleared if the switch reboots before the next save.

If the LPS port is shut down or the network device is disconnected from the port, the LPS table entries for this port are retained, but the source learning MAC address table entries for the same port are automatically cleared. In addition, if an LPS table entry is intentionally cleared from the table, the MAC address for this entry is automatically cleared from the source learning table at the same time.

To view the contents of the LPS table, use the [show port-security](#) command. Refer to the *OmniSwitch 6250 CLI Reference Guide* for more information about this command.

Configuring Learned Port Security

This section describes how to use Command Line Interface (CLI) command to configure Learned Port Security (LPS) on a switch. See the [“Sample Learned Port Security Configuration” on page 3-3](#) for a brief tutorial on configuring LPS.

Configuring LPS involves the following procedures:

- Enabling LPS for one or more switch ports. This procedure is described in [“Enabling/Disabling Learned Port Security” on page 3-7](#).
- Configuring the source learning time window during which MAC addresses are learned. This procedure is described in [“Configuring a Source Learning Time Limit” on page 3-8](#).
- Configuring the maximum number of bridged MAC addresses allowed on an LPS port. This procedure is described in [“Configuring the Number of Bridged MAC Addresses Allowed” on page 3-9](#).
- Configuring the maximum number of filtered MAC addresses allowed on an LPS port. This procedure is describe in [“Configuring the Number of Filtered MAC Addresses Allowed” on page 3-10](#)
- Configuring one or more static authorized MAC addresses. This procedure is described in [“Configuring Authorized MAC Addresses” on page 3-10](#).
- Specifying whether or not an LPS port shuts down all traffic or only restricts traffic when an unauthorized MAC address is received on the port. This procedure is described in [“Selecting the Security Violation Mode” on page 3-11](#).

Enabling/Disabling Learned Port Security

By default, LPS is disabled on all switch ports. To enable LPS on a port, use the **port-security** command. For example, the following command enables LPS on port 1 of slot 4:

```
-> port-security 4/1 enable
```

To enable LPS on multiple ports, specify a range of ports or multiple slots. For example:

```
-> port-security 4/1-5 enable  
-> port-security 5/12-20 6/10-15 enable
```

Note that when LPS is enabled on an active port, all MAC addresses learned on that port prior to the time LPS was enabled are cleared from the source learning MAC address table.

To disable LPS on a port, use the **port-security** command with the **disable** parameter. For example, the following command disables LPS on a range of ports:

```
-> port-security 5/21-24 6/1-4 disable
```

To disable all the LPS ports on a chassis, use the **port-security chassis disable** command, as shown:

```
-> port-security chassis disable
```

When LPS is disabled on a port, MAC address entries for that port are retained in the LPS table. The next time LPS is enabled on the port, the same LPS table entries are again active. If there is a switch reboot before the switch configuration is saved, however, dynamic MAC address entries are discarded from the table.

Use the **no** form of this command to remove LPS *and* clear all entries (configured and dynamic) in the LPS table for the specified port. For example:

```
-> no port-security 5/10
```

After LPS is removed, all the dynamic and static MAC addresses will be flushed and the learning of new MAC addresses will be enabled.

Configuring a Source Learning Time Limit

By default, the source learning time limit is disabled. Use the **port-security shutdown** command to set the number of minutes the source learning window is to remain open for LPS ports. While this window is open, source MAC addresses that comply with LPS port restrictions are authorized for learning on the related LPS port. The following actions trigger the start of the source learning timer:

- The **port-security shutdown** command. Each time this command is issued, the timer restarts even if a current window is still open or a previous window has expired.
- Switch reboot with a **port-security shutdown** command entry saved in the **boot.cfg** file.

The LPS source learning time limit is a switch-wide parameter that applies to all LPS enabled ports, not just one or a group of LPS ports. The following command example sets the time limit value to 30 minutes:

```
-> port-security shutdown time 30
```

Once the time limit value expires, source learning of any new dynamic MAC addresses is stopped on all LPS ports even if the number of addresses learned does not exceed the maximum allowed.

Note. The LPS source learning time window has a higher priority over the maximum number of MAC addresses allowed. Therefore, if the learning interval expires before the port has learned the maximum MAC addresses allowed, the port will *not* learn anymore MAC addresses.

When the source learning time window expires, all the dynamic MAC addresses learned on the LPS ports start to age out. To prevent this, all dynamic MAC addresses must be converted to static MAC addresses. The **convert-to-static** parameter used with the **port-security shutdown** command enables or disables the conversion of dynamic MAC addresses to static MAC addresses on LPS ports when the source learning time window expires.

To enable the conversion of dynamic MAC addresses to static MAC addresses on LPS ports when the source learning time window expires, use the **port-security shutdown** command with the **convert-to-static** parameter, as shown:

```
-> port-security shutdown 30 convert-to-static enable
```

To disable the conversion of dynamic MAC addresses to static MAC addresses when the source learning time window expires, use the **port-security shutdown** command with the **convert-to-static** parameter, as shown:

```
-> port-security shutdown 30 convert-to-static disable
```

To convert the dynamically learned MAC addresses to static addresses on a specific LPS port at any time irrespective of the source learning time window, use the **port-security convert-to-static** command. For example, to convert the dynamic MAC addresses on port 8 of slot 4 to static ones, enter:

```
-> port-security 4/8 convert-to-static
```

Note. The number of converted static MAC addresses cannot exceed the maximum number of MAC addresses allowed on the LPS ports.

Note. The conversion of dynamic MAC addresses to static ones does not apply to LPS mobile and authenticated ports.

Configuring the Number of Bridged MAC Addresses Allowed

By default, one MAC address is allowed on an LPS port. To change this number, enter **port-security** followed by the port's *slot/port* designation, then **maximum** followed by a number between 1 and 100. For example, the following command sets the maximum number of MAC addresses learned on port 10 of slot 6 to 75:

```
-> port-security 6/10 maximum 75
```

To specify a maximum number of MAC addresses allowed for multiple ports, specify a range of ports or multiple slots. For example:

```
-> port-security 1/10-15 maximum 10  
-> port-security 2/1-5 4/2-8 5/10-14 maximum 25
```

Note that configured MAC addresses count towards the maximum number allowed. For example, if there are 10 configured authorized MAC addresses for an LPS port and the maximum number of addresses allowed is set to 15, then only 5 dynamically learned MAC address are allowed on this port.

If the maximum number of MAC addresses allowed is reached before the switch LPS time limit expires, then all source learning of dynamic *and* configured MAC addresses is stopped on the LPS port.

Configuring the Trap Threshold for Bridged MAC Addresses

The LPS trap threshold value determines how many bridged MAC addresses the port must learn before a trap is sent. Once this value is reached, a trap is sent for every MAC learned thereafter.

By default, when five bridged MAC addresses are learned on an LPS port, the switch sends a trap. To change the trap threshold value, use the **port-security learn-trap-threshold** command. For example:

```
-> port-security learn-trap-threshold 10
```

Sending a trap when this threshold is reached provides notification of newly learned bridged MAC addresses. Trap contents includes identifying information about the MAC, such as the address itself, the corresponding IP address, switch identification, and the slot and port number on which the MAC was learned.

Configuring the Number of Filtered MAC Addresses Allowed

By default, five filtered MAC addresses can be learned on an LPS port. To change this number, enter **port-security** followed by the port's *slot/port* designation, then **max-filtering** followed by a number between 1 and 100. For example, the following command sets the maximum number of filtered MAC addresses learned on port 9 of slot 5 to 18:

```
-> port-security 5/9 max-filtering 18
```

To specify a maximum number of filtered MAC addresses learned on multiple ports, specify a range of ports or multiple slots. For example:

```
-> port-security 5/9-15 max-filtering 10  
-> port-security 1/1-5 7/2-8 2/10-14 max-filtering 25
```

If the maximum number of filtered MAC addresses allowed is reached, either the LPS port is disabled (Shutdown Violation mode) or MAC address learning is disabled (Restrict Violation mode). Under both these modes, SNMP traps are generated and the events are logged in the switch log. For information on configuring the security violation modes, see [“Selecting the Security Violation Mode” on page 3-11](#).

Configuring Authorized MAC Addresses

To configure a single source MAC address entry in the LPS table, enter **port-security** followed by the port's *slot/port* designation, the keyword **mac** followed by a valid MAC address, then **vlan** followed by a VLAN ID. For example, the following command configures a MAC address for port 4 on slot 6 that belongs to VLAN 10:

```
-> port-security 6/4 mac 00:20:da:9f:58:0c vlan 10
```

Note. If a VLAN is not specified, the default VLAN for the port is used.

Use the **no** form of this command to clear configured *and/or* dynamic MAC address entries from the LPS table. For example, the following command removes a MAC address entry for port 4 of slot 6 that belongs to VLAN 10 from the LPS table:

```
-> port-security 6/4 no mac 00:20:da:9f:58:0c vlan 10
```

Note that when a MAC address is cleared from the LPS table, it is automatically cleared from the source learning MAC address table at the same time.

Configuring an Authorized MAC Address Range

By default, each LPS port is set to a range of 00:00:00:00:00:00–ff:ff:ff:ff:ff:ff, which includes all MAC addresses. If this default is not changed, then addresses received on LPS ports are subject only to the source learning time limit and maximum number of MAC addresses allowed restrictions for the port.

To configure a source MAC address range for an LPS port, enter **port-security** followed by the port's *slot/port* designation, then **mac-range** followed by **low** and a MAC address, then **high** and a MAC address. For example, the following command configures a MAC address range for port 1 on slot 4:

```
-> port-security 4/1 mac-range low 00:20:da:00:00:10 high 00:20:da:00:00:50
```

To configure a source MAC address range for multiple ports, specify a range of ports or multiple slots. For example:

```
-> port-security 4/1-5 mac-range low 00:20:da:00:00:10 high 00:20:da:00:00:50
-> port-security 2/1-4 4/5-8 mac-range low 00:20:d0:59:0c:9a high
00:20:d0:59:0c:9f
```

To set the range back to the default values, enter **port-security** followed by the port's *slot/port* designation, then **mac-range**. Leaving off the **low** and **high** MAC addresses will reset the range back to 00:00:00:00:00:00 and ff:ff:ff:ff:ff:ff. For example, the following command sets the authorized MAC address range to the default values for port 12 of slot 4:

```
-> port-security 4/12 mac-range
```

In addition, specifying a low end MAC and a high end MAC is optional. If either one is not specified, the default value is used. For example, the following commands set the authorized MAC address range on the specified ports to 00:da:25:59:0c:10–ff:ff:ff:ff:ff:ff and 00:00:00:00:00:00–00:da:25:00:00:9a:

```
-> port-security 2/8 mac-range low pp:da:25:59:0c
-> port-security 2/10 mac-range high 00:da:25:00:00:9a
```

Refer to the *OmniSwitch 6250 CLI Reference Guide* for more information about this command.

Selecting the Security Violation Mode

By default, the security violation mode for an LPS port is set to **restrict**. In this mode, when an unauthorized MAC address is received on an LPS port, the packet containing the address is blocked. However, all other packets that contain an authorized source MAC address are allowed to forward on the port.

Note that unauthorized source MAC addresses are not learned in the LPS table but are still recorded in the source learning MAC address table with a filtered operational status. This allows the user to view MAC addresses that were attempting unauthorized access to the LPS port.

The other violation mode option is **shutdown**. In this mode, the LPS port is disabled when an unauthorized MAC address is received; all traffic is prevented from forwarding on the port. After a shutdown occurs, a manual reset is required to return the port back to normal operation.

To configure the security violation mode for an LPS port, enter **port-security** followed by the port's *slot/port* designation, then **violation** followed by **restrict** or **shutdown**. For example, the following command selects the shutdown mode for port 1 on slot 4:

```
-> port-security 4/1 violation shutdown
```

To configure the security violation mode for multiple LPS ports, specify a range of ports or multiple slots. For example:

```
-> port-security 4/1-10 violation shutdown
-> port-security 1/10-15 2/1-10 violation restrict
```

Displaying Learned Port Security Information

To display LPS port and table information, use the show commands listed below:

- | | |
|-------------------------------------------|--------------------------------------------------------------------------------------|
| port-security learn-trap-threshold | Displays Learned Port Security (LPS) configuration and table entries. |
| show port-security shutdown | Displays the amount of time during which source learning can occur on all LPS ports. |

For more information about the resulting display from these commands, see the *OmniSwitch 6250 CLI Reference Guide*. An example of the output for the **show port-security** and **show port-security shutdown** commands is also given in [“Sample Learned Port Security Configuration”](#) on page 3-3.

4 Configuring VLANs

In a flat bridged network, a broadcast domain is confined to a single LAN segment or even a specific physical location, such as a department or building floor. In a switch-based network, such as one comprised of Alcatel-Lucent switching systems, a broadcast domain—or *VLAN*—can span multiple physical switches and can include ports from a variety of media types. For example, a single VLAN could span three different switches located in different buildings and include 10/100 Ethernet, Gigabit Ethernet, 802.1q tagged ports and/or a link aggregate of ports.

In This Chapter

This chapter describes how to define and manage VLAN configurations through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- “[Creating/Modifying VLANs](#)” on page 4-5.
- “[Defining VLAN Port Assignments](#)” on page 4-7.
- “[Enabling/Disabling VLAN Mobile Tag Classification](#)” on page 4-9.
- “[Enabling/Disabling Spanning Tree for a VLAN](#)” on page 4-10.
- “[Configuring VLAN Router Interfaces](#)” on page 4-11.
- “[Bridging VLANs Across Multiple Switches](#)” on page 4-12.
- “[Verifying the VLAN Configuration](#)” on page 4-13.

For information about statically and dynamically assigning switch ports to VLANs, see [Chapter 6](#), “[Assigning Ports to VLANs](#).”

For information about defining VLAN rules that allow dynamic assignment of mobile ports to a VLAN, see [Chapter 8](#), “[Defining VLAN Rules](#).”

For information about Spanning Tree, see [Chapter 11](#), “[Configuring Spanning Tree Parameters](#).”

For information about routing, see [Chapter 25](#), “[Configuring IP](#).”

VLAN Specifications

Note that the maximum limit values provided in the following Specifications table are subject to available system resources:

RFCs Supported	2674 - <i>Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions</i>
IEEE Standards Supported	802.1Q - <i>Virtual Bridged Local Area Networks</i> 802.1D - <i>Media Access Control Bridges</i>
Platforms Supported	OmniSwitch 6250 Series
Maximum VLANs per switch	4094
Maximum VLAN port associations (VPA) per switch	32768
Maximum 802.1Q VLAN port associations per switch	2500
Maximum IP router interfaces per switch	128 IP
Maximum IP router interfaces per VLAN	8
Maximum Spanning Tree VLANs per switch	252
Maximum authenticated VLANs per switch	128
MAC Router Mode Supported	Single
CLI Command Prefix Recognition	All VLAN management commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

VLAN Defaults

Parameter Description	Command	Default
VLAN identifier (VLAN ID)	vlan	VLAN 1 predefined on each switch.
VLAN administrative state	vlan	Enabled
VLAN description	vlan name	VLAN identifier (VLAN ID)
VLAN Spanning Tree state	vlan stp	Enabled (Disabled if VLAN count exceeds 254)
VLAN mobile tag status	vlan mobile-tag	Disabled
VLAN IP router interface	ip interface	VLAN 1 router interface.
VLAN port associations	vlan port default	All ports initially associated with default VLAN 1.

Sample VLAN Configuration

The following steps provide a quick tutorial that will create VLAN 255. Also included are steps to define a VLAN description, IP router interface, and static switch port assignments.

Note. Optional. Creating a new VLAN involves specifying a VLAN ID that is not already assigned to an existing VLAN. To determine if a VLAN already exists in the switch configuration, enter **show vlan**. If VLAN 255 does not appear in the **show vlan** output, then it does not exist on the switch. For example:

```
-> show vlan
```

vlan	type	admin	oper	stree			auth	ip	mble	
				1x1	flat				tag	name
1	std	on	on	on	on	off	NA	off	VLAN 1	
2	gvrp	on	on	off	off	off	NA	off	GVRPVLAN 2	
3	ipmv	on	on	off	off	off	NA	off	IPMVVLAN 3	
4	vstk	on	on	on	on	off	NA	off	SVLAN 4	

1 Create VLAN 255 with a description (e.g., Finance IP Network) using the following command:

```
-> vlan 255 name "Finance IP Network"
```

2 Define an IP router interface using the following command to assign an IP host address of 21.0.0.10 to VLAN 255 that will enable routing of VLAN traffic to other subnets:

```
-> ip interface vlan-255 address 21.0.0.10 vlan 255
```

3 Assign switch ports 2 through 4 on slot 3 to VLAN 255 using the following command:

```
-> vlan 255 port default 3/2-4
```

Note. Optional. To verify the VLAN 255 configuration, use the **show vlan** command. For example:

```
-> show vlan 255
Name                : Finance IP Network,
Administrative State: enabled,
Operational State   : disabled,
1x1 Spanning Tree State : enabled,
Flat Spanning Tree State : enabled,
Authentication      : disabled,
IP Router Port       : 21.0.0.10 255.0.0.0 forward e2,
Mobile Tag           : off
```

To verify that ports 3/2-4 were assigned to VLAN 255, use the **show vlan port** command. For example:

```
-> show vlan 255 port
```

port	type	status
3/2	default	inactive
3/3	default	inactive
3/4	default	inactive

VLAN Management Overview

One of the main benefits of using VLANs to segment network traffic, is that VLAN configuration and port assignment is handled through switch software. This eliminates the need to physically change a network device connection or location when adding or removing devices from the VLAN broadcast domain. The VLAN management software handles the following VLAN configuration tasks performed on an Alcatel-Lucent switch:

- Creating or modifying VLANs.
- Assigning or changing default VLAN port associations (VPAs).
- Enabling or disabling VLAN participation in the current Spanning Tree algorithm.
- Enabling or disabling classification of mobile port traffic by 802.1Q tagged VLAN ID.
- Enabling or disabling VLAN authentication.
- Enabling or disabling unique MAC address assignments for each router VLAN defined.
- Displaying VLAN configuration information.

In addition to the above tasks, VLAN management software tracks and reports the following information to other switch software applications:

- VLAN configuration changes, such as adding or deleting VLANs, modifying the status of VLAN properties (e.g., administrative, Spanning Tree, and authentication status), changing the VLAN description, or configuring VLAN router interfaces.
- VLAN port associations triggered by VLAN management and other switch software applications, such as 802.1Q VLAN tagging and dynamic mobile port assignment.
- The VLAN operational state, which is inactive until at least one active switch port is associated with the VLAN.

Creating/Modifying VLANs

The initial configuration for all Alcatel-Lucent switches consists of a default VLAN 1 and all switch ports are initially assigned to this VLAN. When a switching module is added to the switch, the module's physical ports are also assigned to VLAN 1. If additional VLANs are not configured on the switch, then the entire switch is treated as one large broadcast domain. All ports will receive all traffic from all other ports.

Up to 4094 VLANs are supported per switch, including default VLAN 1. In compliance with the IEEE 802.1Q standard, each VLAN is identified by a unique number, referred to as the *VLAN ID*. The user specifies a VLAN ID to create, modify or remove a VLAN and to assign switch ports to a VLAN. When a packet is received on a port, the port's VLAN ID is inserted into the packet. The packet is then bridged to other ports that are assigned to the same VLAN ID. In essence, the VLAN broadcast domain is defined by a collection of ports and packets assigned to its VLAN ID.

The operational status of a VLAN remains inactive until at least one active switch port is assigned to the VLAN. This means that VLAN properties, such as Spanning Tree or router interfaces, also remain inactive. Ports are considered active if they are connected to an active network device. Non-active port assignments are allowed, but do not change the VLAN's operational state.

Ports are either statically or dynamically assigned to VLANs. When a port is assigned to a VLAN, a VLAN port association (VPA) is created and tracked by VLAN management switch software. For more information about VPAs, see [“Defining VLAN Port Assignments” on page 4-7](#) and [Chapter 6, “Assigning Ports to VLANs.”](#)

Adding/Removing a VLAN

To add a VLAN to the switch configuration, enter **vlan** followed by a unique VLAN ID number between 2 and 4094, an optional administrative status, and an optional description. For example, the following command creates VLAN 755 with a description:

```
-> vlan 755 enable name "IP Finance Network"
```

By default, administrative status and Spanning Tree are enabled when the VLAN is created and the VLAN ID is used for the description if one is not specified. Note that quotation marks are required if the description contains multiple words separated by spaces. If the description consists of only one word or multiple words separated by another character, such as a hyphen, then quotes are not required.

You can also specify a range of VLAN IDs with the **vlan** command. Use a hyphen to indicate a contiguous range and a space to separate multiple VLAN ID entries. For example, the following command creates VLANs 10 through 15, 100 through 105, and VLAN 200 on the switch:

```
-> vlan 10-15 100-105 200 name "Marketing Network"
```

To remove a VLAN from the switch configuration, use the **no** form of the **vlan** command.

```
-> no vlan 755
-> no vlan 100-105
-> no vlan 10-15 200
```

When a VLAN is deleted, any router interfaces defined for the VLAN are removed and all VLAN port associations are dropped. For more information about VLAN router interfaces, see [“Configuring VLAN Router Interfaces” on page 4-11](#).

Note that up to 253 Spanning Tree instances per switch are supported in the 1x1 Spanning Tree mode. Since each VLAN with Spanning Tree enabled uses one of these instances, only 253 VLANs can have an active Spanning Tree instance at any given time.

To create more than 253 VLANs on a switch running in the 1x1 Spanning Tree mode, use the **vlan stp disable**, **vlan 1x1 stp disable**, or **vlan flat stp disable** command to create a VLAN with Spanning Tree disabled. See [“Enabling/Disabling Spanning Tree for a VLAN” on page 4-10](#) for more information.

To view a list of VLANs already configured on the switch, use the **show vlan** command. See [“Verifying the VLAN Configuration” on page 4-13](#) for more information.

Enabling/Disabling the VLAN Administrative Status

To enable or disable the administrative status for an existing VLAN, enter **vlan** followed by an existing VLAN ID and either **enable** or **disable**.

```
-> vlan 755 disable
-> vlan 255 enable
```

When the administrative status for a VLAN is disabled, VLAN port assignments are retained but traffic is not forwarded on these ports. If any rules were defined for the VLAN, they are also retained and continue to classify mobile port traffic. See [Chapter 8, “Defining VLAN Rules,”](#) for more information.

Modifying the VLAN Description

To change the description for a VLAN, enter **vlan** followed by an existing VLAN ID and the keyword **name** followed by the new description (up to 32 characters). For example, the following command changes the description for VLAN 455 to “Marketing IP Network”:

```
-> vlan 455 name "Marketing IP Network"
```

Note that quotation marks are required if the description consists of multiple words separated by spaces. If the description consists of only one word or words are separated by another character, such as a hyphen, then quotes are not required. For example,

```
-> vlan 455 name Marketing-IP-Network
```

Defining VLAN Port Assignments

Alcatel-Lucent switches support static and dynamic assignment of physical switch ports to a VLAN. Regardless of how a port is assigned to a VLAN, once the assignment occurs, a VLAN port association (VPA) is created and tracked by VLAN management software on each switch. To view current VLAN port assignments in the switch configuration, use the **show vlan port** command.

Methods for statically assigning ports to VLANs include the following:

- Using the **vlan port default** command to define a new configured default VLAN for both non-mobile (fixed) and mobile ports. (See “[Changing the Default VLAN Assignment for a Port](#)” on page 4-7.)
- Using the **vlan 802.1q** command to define tagged VLANs for non-mobile ports. This method allows the switch to bridge traffic for multiple VLANs over one physical port connection. (See [Chapter 22, “Configuring 802.1Q.”](#))
- Configuring ports as members of a link aggregate that is assigned to a configured default VLAN. (See [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation,”](#) for more information.)

Dynamic assignment applies only to mobile ports. When traffic is received on a mobile port, the packets are classified using one of the following methods to automatically determine VLAN assignment (see [Chapter 6, “Assigning Ports to VLANs,”](#) for more information):

- Packet is tagged with a VLAN ID that matches the ID of another VLAN that has mobile tagging enabled. (See “[Enabling/Disabling VLAN Mobile Tag Classification](#)” on page 4-9.)
- Packet contents matches criteria defined in a VLAN rule. (See “[Configuring VLAN Rule Classification](#)” on page 4-8 and [Chapter 8, “Defining VLAN Rules.”](#))

Changing the Default VLAN Assignment for a Port

To assign a switch port to a new default VLAN, enter **vlan** followed by an existing VLAN ID number, **port default**, then the slot/port designation. For example, the following command assigns port 5 on slot 2 to VLAN 955:

```
-> vlan 955 port default 2/5
```

All ports initially belong to default VLAN 1. When the **vlan port default** command is used, the port’s default VLAN assignment is changed to the specified VLAN. In the above example, VLAN 955 is now the default VLAN for port 5 on slot 2 and this port is no longer associated with VLAN 1.

The **vlan port default** command is also used to change the default VLAN assignment for an aggregate of ports. The link aggregate control number is specified instead of a slot and port. For example, the following command assigns link aggregate 10 to VLAN 755:

```
-> vlan 755 port default 10
```

For more information about configuring an aggregate of ports, see [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Use the **no** form of the **vlan port default** command to remove a default VPA. When this is done, VLAN 1 is restored as the port’s default VLAN.

```
-> vlan 955 no port default 2/5
```

Configuring Dynamic VLAN Port Assignment

Configuring the switch to allow dynamic VLAN port assignment requires the following steps:

- 1 Use the **vlan port mobile** command to enable mobility on switch ports that will participate in dynamic VLAN assignment. See [Chapter 6, “Assigning Ports to VLANs,”](#) for detailed procedures.
- 2 Enable/disable mobile port properties that determine mobile port behavior. See [Chapter 6, “Assigning Ports to VLANs,”](#) for detailed procedures.
- 3 Create VLANs that will receive and forward mobile port traffic. See [“Adding/Removing a VLAN” on page 4-5](#) for more information.
- 4 Configure the method of traffic classification (VLAN rules or tagged VLAN ID) that will trigger dynamic assignment of mobile ports to the VLANs created in Step 3. See [“Configuring VLAN Rule Classification” on page 4-8](#) and [“Enabling/Disabling VLAN Mobile Tag Classification” on page 4-9](#).

Once the above configuration steps are completed, dynamic VLAN assignment occurs when a device connected to a mobile port starts to send traffic. This traffic is examined by switch software to determine which VLAN should carry the traffic based on the type of classification, if any, defined for a particular VLAN.

Note that VLAN mobile tag classification takes precedence over VLAN rule classification. If a mobile port receives traffic that matches a VLAN rule and also has an 802.1Q VLAN ID tag for a VLAN with mobile tagging enabled, the port is dynamically assigned to the mobile tag VLAN and not the matching rule VLAN.

See [Chapter 6, “Assigning Ports to VLANs,”](#) and [Chapter 8, “Defining VLAN Rules,”](#) for more information and examples of dynamic VLAN port assignment.

Configuring VLAN Rule Classification

VLAN rule classification triggers dynamic VLAN port assignment when traffic received on a mobile port matches the criteria defined in a VLAN rule. Different rule types are available for classifying different types of network device traffic. It is possible to define multiple rules for one VLAN and rules for multiple VLANs.

The following table provides a list of commands used to define the various types of VLAN rules. For more detailed information about rule criteria and classification, see [Chapter 8, “Defining VLAN Rules.”](#)

Rule Types	Command
DHCP	vlan dhcp mac vlan dhcp mac range vlan dhcp port vlan dhcp generic
MAC address	vlan mac vlan mac range
Network address	vlan ip vlan protocol
Protocol	vlan protocol
Port	vlan port

Enabling/Disabling VLAN Mobile Tag Classification

Use the **vlan mobile-tag** command to enable or disable the classification of mobile port packets based on 802.1Q VLAN ID tag. For example, the following commands enable the mobile tag attribute for VLAN 1525 and disable it for VLAN 224:

```
-> vlan 1525 mobile-tag enable
-> vlan 224 mobile-tag disable
```

If a mobile port that is statically assigned to VLAN 10 receives an 802.1Q tagged packet with a VLAN ID of 1525, the port and packet are dynamically assigned to VLAN 1525. In this case, the mobile port now has a VLAN port association defined for VLAN 10 and for VLAN 1525. If a mobile port, however, receives a tagged packet containing a VLAN ID tag of 224, the packet is discarded because the VLAN mobile tag classification attribute is disabled on VLAN 224.

In essence, the VLAN mobile tag attribute provides a dynamic 802.1Q tagging capability. Mobile ports can now receive and process 802.1Q tagged packets destined for a VLAN that has this attribute enabled. This feature also allows the dynamic assignment of mobile ports to more than one VLAN at the same time, as discussed in the above example.

VLAN mobile tagging differs from 802.1Q tagging as follows:

VLAN Mobile Tag	802.1Q Tag
Allows mobile ports to receive 802.1Q tagged packets.	Not supported on mobile ports.
Enabled on the VLAN that will receive tagged mobile port traffic.	Enabled on fixed ports; tags port traffic for destination VLAN.
Triggers dynamic assignment of tagged mobile port traffic to one or more VLANs.	Statically assigns (tags) fixed ports to one or more VLANs.

If 802.1Q tagging is required on a fixed (non-mobile) port, then the **vlan 802.1q** command is still used to statically tag VLANs for the port. See [Chapter 22, “Configuring 802.1Q,”](#) for more information.

Enabling/Disabling Spanning Tree for a VLAN

The spanning tree operating mode set for the switch determines how VLAN ports are evaluated to identify redundant data paths. If the Spanning Tree switch operating mode is set to *flat*, then VLAN port connections are checked against other VLAN port connections for redundant data paths. Note that the single flat mode STP instance is referred to as *instance 1* or the CIST (Common and Internal Spanning Tree) instance, depending on which STP protocol is active.

In the flat mode, if STP instance 1 or the CIST instance is disabled, then it is disabled for all configured VLANs. However, disabling STP on an individual VLAN will exclude only that VLAN's ports from the flat STP algorithm.

If the Spanning Tree operating mode is set to *1x1*, there is a single Spanning Tree instance for each VLAN broadcast domain. Enabling or disabling STP on a VLAN in this mode will include or exclude the VLAN from the 1x1 STP algorithm.

The **vlan stp** command is used to enable/disable a Spanning Tree instance for an existing VLAN. In the following examples, Spanning Tree is disabled on VLAN 255 and enabled on VLAN 755:

```
-> vlan 255 stp disable
-> vlan 755 stp enable
```

Note the following when using the **vlan stp** command. For more information about the **vlan stp** command, see the *OmniSwitch 6250 CLI Reference Guide*:

- If the VLAN ID specified with this command is that of a VLAN that does not exist, the VLAN is automatically created.
- This command configures the VLAN STP status for both the 1x1 and flat Spanning Tree modes. Using the **1x1** or **flat** parameter with this command, configures the STP status only for the mode specified by the parameter.
- Up to 253 Spanning Tree instances per switch are supported in the 1x1 Spanning Tree mode. Since each VLAN with Spanning Tree enabled uses one of these instances, only 253 VLANs can have an active Spanning Tree instance at any given time.
- To create more than 253 VLANs on a switch running in the 1x1 Spanning Tree mode, use the **vlan stp disable**, **vlan 1x1 stp disable**, or **vlan flat stp disable** form of this command to create a VLAN with Spanning Tree disabled.

STP does not become operationally active on a VLAN unless the VLAN is operationally active, which occurs when at least one active port is assigned to the VLAN. Also, STP is enabled/disabled on individual ports. So even if STP is enabled for the VLAN, a port assigned to that VLAN must also have STP enabled. See [Chapter 11, "Configuring Spanning Tree Parameters."](#)

Configuring VLAN Router Interfaces

Network device traffic is bridged (switched) at the Layer 2 level between ports that are assigned to the same VLAN. However, if a device needs to communicate with another device that belongs to a different VLAN, then Layer 3 routing is necessary to transmit traffic between the VLANs. Bridging makes the decision on where to forward packets based on the packet's destination MAC address; routing makes the decision on where to forward packets based on the packet's IP network address (e.g., IP - 21.0.0.10).

Alcatel-Lucent switches support routing of IP traffic. A VLAN is available for routing when at least one router interface is defined for that VLAN and at least one active port is associated with the VLAN. Up to eight IP interfaces can be configured for each VLAN. The maximum number of IP interfaces allowed for the entire switch is 4094.

If a VLAN does not have a router interface, the ports associated with that VLAN are in essence firewalled from other VLANs. For information about how to configure router interfaces, see [Chapter 25, "Configuring IP."](#)

What is Single MAC Router Mode?

The switch operates only in single MAC router mode. In this mode, each router VLAN is assigned the same MAC address, which is the base chassis MAC address for the switch. This eliminates the need to allocate additional MAC addresses if more than 32 router VLANs are defined. The number of router VLANs allowed then is based on the IP interface configuration. See "[Configuring VLAN Router Interfaces](#)" on page 4-11 for more information.

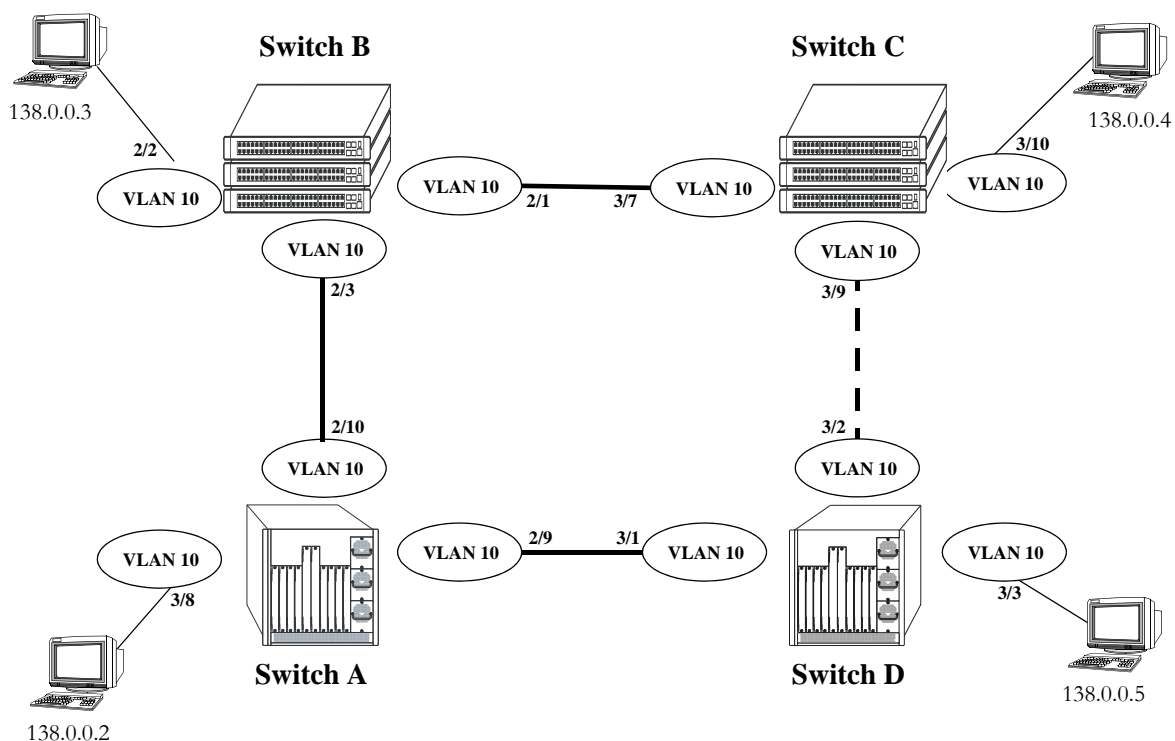
To determine the total number of VLANs configured on the switch, and the number of VLANs with IP router interfaces configured, use the `show vlan router mac status` command. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Bridging VLANs Across Multiple Switches

To create a VLAN *bridging domain* that extends across multiple switches:

- 1 Create a VLAN on each switch with the same VLAN ID number (e.g., VLAN 10).
- 2 If using mobile ports for end user device connections, define VLAN rules that will classify mobile port traffic into the VLAN created in Step 1.
- 3 On each switch, assign the ports that will provide connections to other switches to the VLAN created in Step 1.
- 4 On each switch, assign the ports that will provide connections to end user devices (e.g., workstations) to the VLAN created in Step 1. (If using mobile ports, this step will occur automatically when the device connected to the mobile port starts to send traffic.)
- 5 Connect switches and end user devices to the assigned ports.

The following diagram shows the physical configuration of an example VLAN bridging domain:

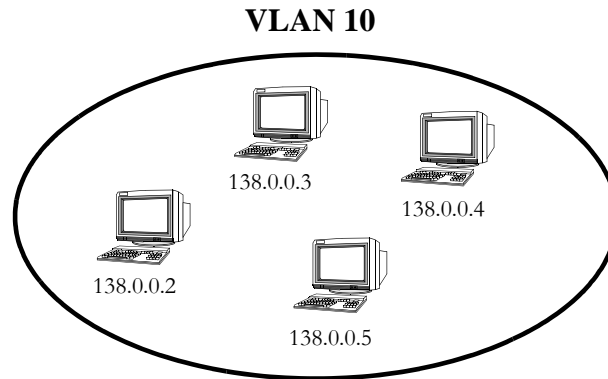


VLAN Bridging Domain: Physical Configuration

In the above diagram, VLAN 10 exists on all four switches and the connection ports between these switches are assigned to VLAN 10. The workstations can communicate with each other because the ports to which they are connected are also assigned to VLAN 10. It is important to note that connection cables do not have to connect to the same port on each switch. The key is that the port must belong to the same VLAN on each switch. To carry multiple VLANs between switches across a single physical connection cable, use the 802.1Q tagging feature (see [Chapter 22, "Configuring 802.1Q"](#)).

The connection between Switch C and D is shown with a broken line because the ports that provide this connection are in a blocking state. Spanning Tree is active by default on all switches, VLANs and ports. The Spanning Tree algorithm determined that if all connections between switches were active, a network loop would exist that could cause unnecessary broadcast traffic on the network. The path between Switch C and D was shut down to avoid such a loop. See [Chapter 11, “Configuring Spanning Tree Parameters,”](#) for information about how Spanning Tree configures network topologies that are loop free.

The following diagram shows the same bridging domain example as seen by the end user workstations. Because traffic between these workstations is *bridged* across physical switch connections within the VLAN 10 domain, the workstations are basically unaware that the switches even exist. Each workstation believes that the others are all part of the same VLAN, even though they are physically connected to different switches.



VLAN Bridging Domain: Logical View

Creating a VLAN bridging domain across multiple switches and/or stacks of switches allows VLAN members to communicate with each other, even if they are not connected to the same physical switch. This is how a logical grouping of users can traverse a physical network setup without routing and is one of the many benefits of using VLANs.

Verifying the VLAN Configuration

To display information about the VLAN configuration for a single switch or a stack of switches, use the show commands listed below:

show vlan	Displays a list of all VLANs configured on the switch and the status of related VLAN properties (e.g., admin and Spanning Tree status and router port definitions).
show vlan port	Displays a list of VLAN port assignments.
show ip interface	Displays VLAN IP router interface information.
show vlan router mac status	Displays the current MAC router operating mode (single or multiple) and VLAN router port statistics.

For more information about the resulting displays from these commands, see the *OmniSwitch 6250 CLI Reference Guide*. An example of the output for the **show vlan** and **show vlan port** commands is also given in [“Sample VLAN Configuration”](#) on page 4-3.

5 Configuring GVRP

The GARP VLAN Registration Protocol (GVRP) facilitates in controlling virtual local area networks (VLANs) in a large network. It is an application of Generic Attribute Registration Protocol (GARP) and provides VLAN registration service. GVRP enables devices to dynamically learn their VLAN memberships.

GVRP is compliant with 802.1Q standard. It dynamically learns and propagates VLAN membership information across a bridged network. GVRP dynamically maintains and updates the registration and de-registration of VLANs and prunes unnecessary broadcast and unicast traffic. Through the propagation of GVRP information, a device is continuously able to update its knowledge on the set of VLANs that currently have active nodes and on the ports through which those nodes can be reached.

In This Chapter

This chapter describes the basic components of GVRP and their configuration through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling GVRP on [page 5-7](#).
- Enabling Transparent Switching on [page 5-8](#).
- Configuring Maximum Number of VLANs on [page 5-8](#).
- Configuring GVRP Registration on [page 5-9](#).
- Configuring GVRP Applicant Mode on [page 5-10](#).
- Modifying GVRP Timers on [page 5-10](#).
- Restricting VLAN Registration on [page 5-11](#).
- Restricting Static VLAN Registration on [page 5-12](#).
- Restricting VLAN Advertisements on [page 5-12](#).

GVRP Specifications

IEEE Standards Supported	IEEE Std. 802.1D - 2004, Media Access Control (MAC) Bridges IEEE Draft Std. P802.1Q-REV/D5.0
Platforms Supported	OmniSwitch 6250 Series
Maximum GVRP VLANs	256

GVRP Defaults

The following table lists the defaults for GVRP configuration:

Parameter Description	Command	Default Value/Comments
Global status of GVRP	<code>gvrp</code>	disabled
Status of GVRP on specified port	<code>gvrp port</code>	disabled
Transparent switching	<code>gvrp transparent switching</code>	disabled
Maximum number of VLANs	<code>gvrp maximum vlan</code>	1024
Registration mode of the port	<code>gvrp registration</code>	normal
Applicant mode of the port	<code>gvrp applicant</code>	participant
Timer value for Join timer, Leave timer, or LeaveAll timer	<code>gvrp timer</code>	Join timer value: 600 ms Leave timer value: 1800 ms LeaveAll timer value: 30000 ms
Restrict dynamic VLAN registration	<code>gvrp restrict-vlan-registration</code>	not restricted
Restrict VLAN advertisement	<code>gvrp restrict-vlan-advertisement</code>	not restricted
Restrict static VLAN registration	<code>gvrp static-vlan restrict</code>	not restricted
Maximum VLANs learned through GVRP	<code>gvrp maximum vlan</code>	256

GARP Overview

GARP was introduced to avoid manual configuration of devices and applications in a large network. It enables dynamic configuration of devices and applications in a network. It also provides a generic framework whereby devices in a bridged LAN can register and de-register attribute values, such as VLAN identifiers, with each other. These attributes are propagated through devices in the bridged LAN. GARP consists of:

GARP Information Declaration (GID)—The part of GARP that generates data from the switch.

GARP Information Propagation (GIP)—The part of GARP that distributes data to different switches.

A GARP applicant may or may not choose to actively participate in declaring and registering an attribute value. By declaring an attribute, a GARP applicant indicates to other applicants that it is either associated with the attribute or it is interested to know about the other applicants associated with that attribute. A GARP applicant that declares attributes is referred to as an active member. A passive member is an applicant interested in an attribute but will not initiate GARP PDUs when it is aware that other applicants have also registered the attribute.

The following messages are used in GARP:

JoinIn and JoinEmpty—Used by an applicant (including itself) associated with an attribute. Receiving JoinIn messages from other applicants or transmitting JoinEmpty messages enables an applicant to register the attribute.

LeaveIn and LeaveEmpty—Used by an applicant to withdraw its declaration when it is no more associated with an attribute.

LeaveAll—Used for periodic declarations and registration maintenance. An applicant periodically sends LeaveAll messages, which enable other applicants to indicate their attributes' registered states.

These messages indicate the current state of the sender applicant device to other GARP applicant devices. With this information, these GARP applicant devices can modify their behavior associated with the attribute (declare and withdraw).

GVRP Overview

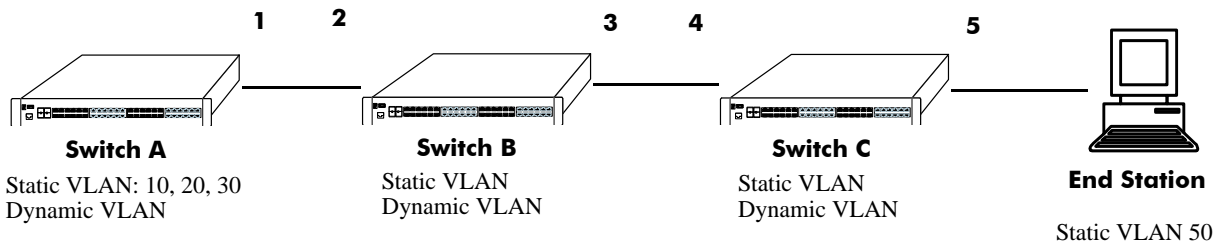
GVRP, an application of GARP, is designed to propagate VLAN information from device to device. With GVRP, a single switch is manually configured with all the desired VLANs for the network, and all the other switches on the network learn those VLANs dynamically. An end station can be plugged into a switch and be connected to its desired VLAN. However, end stations need GVRP-aware Network Interface Cards (NIC) to make use of GVRP.

GVRP sends information encapsulated in an Ethernet frame to a specific MAC address (01:80:C2:00:00:21). Based on the received registration information (Join message of GARP), VLAN information is learned on a system. GVRP enables new dynamic VLANs on a device or dynamically registers a port to an existing VLAN. In effect, based on the received registration information of a VLAN, the port becomes associated with that VLAN. Similarly, whenever de-registration information is received for a VLAN (Leave message of GARP) on a particular port, the association of that VLAN with the port may get deleted.

A GVRP-enabled port sends GVRP PDUs advertising the VLAN. Other GVRP-aware ports receiving advertisements over a link can dynamically join the advertised VLAN. All ports of a dynamic VLAN operate as tagged ports for that VLAN. Also, a GVRP-enabled port can forward an advertisement for a

VLAN it learned about from other ports on the same switch. However, that forwarding port does not join that VLAN until an advertisement for that VLAN is received on that port.

The following illustration shows dynamic VLAN advertisements:



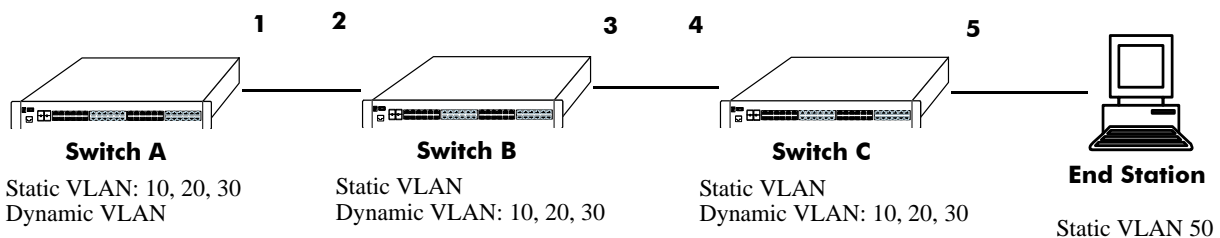
Initial Configuration of GVRP

Switch A has 3 VLANs configured as static VLANs (10, 20, and 30). Other switches on the same network will learn these 3 VLANs as dynamic VLANs. Also, the end station connected on port 5 is statically configured for VLAN 50. Port 1 on Switch A is manually configured for VLANs 10, 20, and 30. Hence, as the diagram above shows,

- 1 Port 1 on Switch A advertises VLAN IDs (VIDs) 10, 20, and 30.
- 2 Port 2 on Switch B receives the advertisements. VLANs 10, 20, and 30 are created as dynamic VLANs on this switch and Port 2 becomes a member of VLANs 10, 20, and 30.
- 3 Port 3 on Switch B is triggered to advertise VLANs 10, 20, and 30, but does not become a member of these VLANs.
- 4 Port 4 on Switch C receives the advertisements. VLANs 10, 20, and 30 are created as dynamic VLANs on this switch and Port 4 becomes a member of VLANs 10, 20, and 30.
- 5 Port 5 advertises VLANs 10, 20, and 30, but this port is not a member of these VLANs.

Note. Default VLAN (VLAN 1) exists on all switches, but it is not considered here.

The above sequence of advertisements and registration of VLANs results in the following configuration:



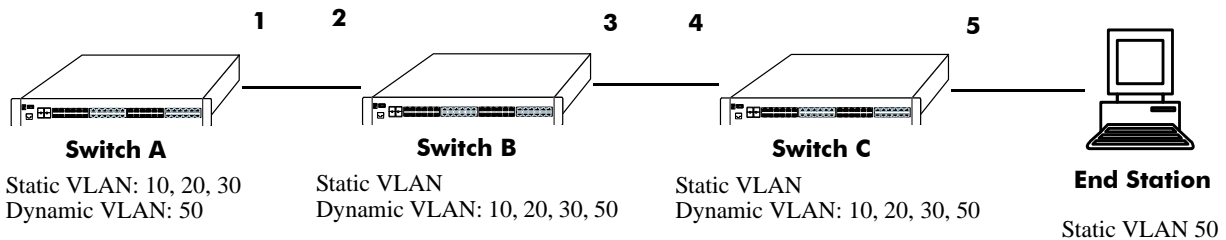
Dynamic Learning of VLANs 10, 20, and 30

Here, the end station advertises itself as a member of VLAN 50. As the above diagram shows,

- 1 Port 5 receives the advertisement and Switch C creates VLAN 50 as a dynamic VLAN. Port 5 of Switch C becomes a member of VLAN 50.
- 2 Port 4 advertises VLAN 50, but is not a member of VLAN 50.

- 3** Port 3 of Switch B receives the advertisement, Switch B creates the dynamic VLAN 50, and Port 3 becomes a member of VLAN 50.
- 4** Port 2 advertises VLAN 50, but is not a member of this VLAN.
- 5** Port 1 on Switch A receives the advertisement, creates dynamic VLAN 50. Port 1 becomes a member of VLAN 50.

The resulting configuration is depicted below:



Dynamic Learning of VLAN 50

Note. Every port on a switch is not a member of all the VLANs. Only those ports that receive the advertisement become members of the VLAN being advertised.

Quick Steps for Configuring GVRP

- 1** Create a VLAN using the **vlan** command. For example:


```
-> vlan 5 name "vlan-7"
```
- 2** Assign a port to the VLAN using the **vlan port default** command. For example:


```
-> vlan 5 port default 3/2
```
- 3** Propagate the VLAN out of the assigned port using the **vlan 802.1q** command. For example, the following command propagates VLAN 5 out of port 3/2:


```
-> vlan 5 802.1q 3/2
```
- 4** Enable GVRP globally on the switch by using the **gvrp** command.


```
-> gvrp
```
- 5** Enable GVRP on the port by using the **gvrp port** command. For example, the following command enables GVRP on port 3/2 of the switch:


```
-> gvrp port 3/2
```
- 6** Restrict a port from becoming a member of the statically created VLAN by using the **gvrp static-vlan restrict** command. For example, the following command restricts port 3/5 from becoming a member of static VLAN 10:


```
-> gvrp static-vlan restrict port 3/5 10
```

7 To view the global configuration details of the router, enter the **show gvrp configuration** command. The globally configured details will be displayed as shown:

```
-> show gvrp configuration

GVRP Enabled           : yes,
Transparent Switching Enabled : no,
Maximum VLAN Limit    : 256
```

8 To view GVRP configuration for a specific port, enter the **show gvrp configuration linkagg/port** command. The configuration details of the particular port will be displayed as shown:

```
-> show gvrp configuration port 1/21

Port 1/21:
GVRP Enabled           : yes,
Registrar Mode         : normal,
Applicant Mode         : participant,
Join Timer (msec)      : 600,
Leave Timer (msec)      : 1800,
LeaveAll Timer (msec)   : 30000,
Legacy Bpdu            : disabled

VLAN Memberships:
VLAN Id      Static      Restricted  Restricted
             Registration Registration Applicant
-----+-----+-----+-----
      1      LEARN      FALSE      FALSE
      2      LEARN      FALSE      FALSE
     11      LEARN      FALSE      FALSE
     12      LEARN      FALSE      FALSE
     13      LEARN      FALSE      FALSE
     14      LEARN      FALSE      FALSE
     15      LEARN      FALSE      FALSE
     16      LEARN      FALSE      FALSE
     17      LEARN      FALSE      FALSE
     18      LEARN      FALSE      FALSE
     19      LEARN      FALSE      FALSE
     20      LEARN      FALSE      FALSE
     51      RESTRICT   FALSE      FALSE
     52      RESTRICT   FALSE      FALSE
     53      LEARN      TRUE       FALSE
     54      LEARN      TRUE       FALSE
     55      LEARN      FALSE      TRUE
     56      LEARN      FALSE      TRUE
     57      LEARN      FALSE      FALSE
     58      LEARN      FALSE      FALSE
     59      LEARN      FALSE      FALSE
     60      LEARN      FALSE      FALSE
```

Configuring GVRP

This section describes how to configure GVRP using Alcatel-Lucent's Command Line Interface (CLI) commands.

Enabling GVRP

GVRP is used primarily to prune unnecessary broadcast and unknown unicast traffic, and dynamically create and manage VLANs. GVRP has to be globally enabled on a switch before it can start forwarding GVRP frames.

To enable GVRP globally on the switch, enter the **gvrp** command at the CLI prompt as shown:

```
-> gvrp
```

To disable GVRP globally on the switch, use the **no** form of the **gvrp** command as shown:

```
-> no gvrp
```

Note. Disabling GVRP globally will lead to the deletion of all learned VLANs.

GVRP can be enabled on ports regardless of whether it is globally enabled or not. However, for the port to become an active participant, you should enable GVRP globally on the switch. By default, GVRP is disabled on the ports. To enable GVRP on a specified port, use the **gvrp port** command.

For example, to enable GVRP on port 2 of slot 1, enter:

```
-> gvrp port 1/2
```

Similarly, to enable GVRP on aggregate group 2, enter:

```
-> gvrp linkagg 2
```

To disable GVRP on a specific port, use the **no** form of the command as shown:

```
-> no gvrp port 1/2
```

Note. GVRP can be configured only on fixed, 802.1 Q and aggregate ports. It cannot be configured on mirror, aggregable, mobile, and MSTI Trunking ports.

Enabling Transparent Switching

A switch in the GVRP transparent mode floods GVRP frames to other switches transparently when GVRP is globally disabled on the switch. However, the switch does not advertise or synchronize its VLAN configuration based on received VLAN advertisements. By default, transparent switching is disabled on the switch.

Note. If GVRP is globally enabled on a switch, transparent switching will have no effect on the switch.

You can configure the switch to propagate GVRP frames transparently using the **gvrp transparent switching** command, as shown:

```
-> gvrp transparent switching
```

Use the **no** form of this command to disable the transparent switching capability of the switch. For example:

```
-> no gvrp transparent switching
```

Note. When both GVRP and GVRP transparent switching are globally disabled, the switch will discard the GVRP frames.

Configuring the Maximum Number of VLANs

A switch can create dynamic VLANs using GVRP. By default, the maximum number of dynamic VLANs that can be created using GVRP is 1024. If the VLAN limit to be set is less than the current number of dynamically learned VLANs, then the new configuration will take effect only after the GVRP is disabled and enabled again on the switch. If this operation is not done, the VLANs learned earlier will be maintained. To modify the maximum number of dynamic VLANs the switch is allowed to create, use the **gvrp maximum vlan** command as shown:

```
-> gvrp maximum vlan 150
```

Here, the number of dynamic VLANs the switch can create is set to a maximum of 150.

Note. A maximum of 4094 dynamic VLANs can be created using GVRP.

These dynamically created VLANs do not support the following operations:

- Authentication
- IP routing
- Configuring default VLAN on any port
- Enabling/Disabling classification of tagged packets received on mobile ports (vlan mobile-tag)

Configuring GVRP Registration

GVRP allows a port to register and de-register both static and dynamic VLANs. Every device has a list of all the switches and end stations that can be reached at any given time. When an attribute for a device is registered or de-registered, the set of reachable switches and end stations, also called participants, is modified. Data frames are propagated only to registered devices. This prevents attempts to send data to devices that are not reachable.

The following sections describe GVRP registration on switches:

Setting GVRP Normal Registration

The normal registration mode allows dynamic creation, registration, and de-registration of VLANs on a device. The normal mode is the default registration mode.

To configure a port in normal mode, use the **gvrp registration** command. For example, to configure port 2 of slot 3 in normal mode, enter the following:

```
-> gvrp registration normal port 3/2
```

To view the registration mode of the port, use the **show gvrp configuration linkagg/port** command. For example:

```
-> show gvrp configuration port 3/2
```

Setting GVRP Fixed Registration

The fixed registration mode allows only manual registration of the VLANs and prevents dynamic or static de-registration of VLANs on the port.

To configure a port to fixed mode, use the **gvrp registration** command. For example, to configure port 2 of slot 3 to fixed mode, enter the following:

```
-> gvrp registration fixed port 3/2
```

To view the registration mode of the port, enter the following:

```
-> show gvrp configuration port 3/2
```

Note. The registration mode for the default VLANs of all the ports in the switch will be set to fixed.

Setting GVRP Forbidden Registration

The forbidden registration mode prevents any VLAN registration or de-registration. If dynamic VLANs previously created are present, they must be de-registered.

To configure a port to forbidden mode, use the **gvrp registration** command. For example, to configure port 2 of slot 3 to forbidden mode, enter the following:

```
-> gvrp registration forbidden port 3/2
```

To view the registration mode of the port, use the **show gvrp configuration linkagg/port** command. For example, to view the mode of port 1/21, enter the following:

```
-> show gvrp configuration port 3/2
```

The GVRP registration mode of the port can be set to default value by using the **no** form of **gvrp registration** command.

To set the GVRP registration mode of port 3/2 to default mode (normal mode) enter the following command:

```
-> no gvrp registration port 3/2
```

Configuring the GVRP Applicant Mode

The GVRP applicant mode determines whether or not GVRP PDU exchanges are allowed on a port, depending on the Spanning Tree state of the port. This mode can be configured to be **participant**, **non-participant** or **active**. By default, the port is in the participant mode.

To prevent undesirable Spanning Tree Protocol topology reconfiguration on a port, configure the GVRP applicant mode as active. Ports in the GVRP active applicant state send GVRP VLAN declarations even when they are in the STP blocking state. This prevents the STP bridge protocol data units (BPDUs) from being pruned from the other ports.

To set the applicant mode of a port to active, use the **gvrp applicant** command. For example, to set the applicant mode of port 3/2 to active, enter the following:

```
-> gvrp applicant active port 3/2
```

When a port is set to participant mode, GVRP protocol exchanges are allowed only if the port is set to the STP forwarding state.

To set the applicant mode of port 3/2 to participant mode, enter the following:

```
-> gvrp applicant participant port 3/2
```

When a port is set to non-participant mode, GVRP PDUs are not sent through the STP forwarding and blocking ports.

To set the applicant mode of port 3/2 to non-participant mode, enter the following:

```
-> gvrp applicant non-participant port 3/2
```

The applicant mode of the port can be set to the default value by using the **no** form of the **gvrp applicant** command. To set the GVRP applicant mode of port 3/2 to the default mode (participant mode), enter the following command:

```
-> no gvrp applicant port 3/2
```

Modifying GVRP timers

GVRP timers control the timing of dynamic VLAN membership updates to connected devices. The following are the various timers in GVRP:

- **Join** timer—The maximum time a GVRP instance waits before making declaration for VLANs.
- **Leave** timer—The wait time taken to remove the port from the VLAN after receiving a Leave message on that port.
- **LeaveAll** timer—The time a GVRP instance takes to generate LeaveAll messages. The LeaveAll message instructs the port to modify the GVRP state of all its VLANs to **Leave**.

The default values of the Join, Leave, and LeaveAll timers are 200 ms, 600 ms, and 10000 ms, respectively.

When you set the timer values, the value for the Leave timer should be greater than or equal to thrice the Join timer value (**Leave** ≥ **Join** * 3). The LeaveAll timer value must be greater than the Leave timer value (**LeaveAll** > **Leave**). If you attempt to set a timer value that does not adhere to these rules, an error message will be displayed.

For example, if you set the Leave timer to 900 ms and attempt to configure the Join timer to 450 ms, an error is returned. You need to set the Leave timer to at least 1350 ms and then set the Join timer to 450 ms.

To modify the Join timer value, use the **gvrp timer** command. For example, to modify the Join timer value of port 3/2, enter the following:

```
-> gvrp timer join 400 port 3/2
```

The Join timer value of port 3/2 is now set to 400 ms.

To set the Join timer to the default value, use the **no** form of the command as shown:

```
-> no gvrp timer join port 3/2
```

To set the Leave timer value of port 3/2 to 1200 ms, enter the command as shown:

```
-> gvrp timer leave 1200 port 3/2
```

To set the LeaveAll timer of port 3/2 to 1400 ms, enter the command as shown:

```
-> gvrp timer leaveall 1200 port 3/2
```

To view the timer value assigned to a particular port, use the **show gvrp timer** command. For example, to view the timer value assigned to port 1/21, enter the command as shown:

```
-> show gvrp configuration port 1/21
```

Note. Set the same GVRP timer value on all the connected devices.

Restricting VLAN Registration

Restricted VLAN registration restricts GVRP from dynamically registering specific VLAN(s) on a switch. It decides whether VLANs can be dynamically created on a device or only be mapped to the ports (if the VLANs are already statically created on the device).

By default, the dynamic VLAN registrations are not restricted and the VLAN can either be created on the device or mapped to another port.

To restrict a VLAN from being dynamically learned on the device, you can configure the dynamic VLAN registrations by using the **gvrp restrict-vlan-registration** command as shown:

```
-> gvrp restrict-vlan-registration port 3/1 4
```

Here, VLAN 4 cannot be learned by the device dynamically. However, if the VLAN already exists on the device as a static VLAN, it can be mapped to the receiving port.

To allow dynamic VLAN registrations on the port, use the **no** form of the [gvrp restrict-vlan-registration](#) command as shown:

```
-> no gvrp restrict-vlan-registration port 3/1 4
```

Restricting Static VLAN Registration

Ports can be exempted from becoming members of statically created VLANs. To restrict a port from becoming a member of a statically configured VLAN, use the [gvrp static-vlan restrict](#) command as shown:

```
-> gvrp static-vlan restrict port 1/2 5
```

Note. This command does not apply to dynamic VLANs.

Here, the port 1/2 is restricted from becoming a GVRP member of VLAN 5.

To restrict a port from becoming a member of a range of statically created VLANs, enter the [gvrp static-vlan restrict](#) command as shown:

```
-> gvrp static-vlan restrict port 1/2 5-9
```

Here, port 1/2 is restricted from becoming a GVRP member of VLANs 5 to 9.

A port can be allowed to become a member of statically created VLANs using the **no** form of the [gvrp static-vlan restrict](#) command. To allow port 3/1 to become a member of a statically created VLAN, enter the command as shown:

```
-> no gvrp static-vlan restrict 3/1
```

Restricting VLAN Advertisement

VLANs learned by a switch through GVRP can either be propagated to other switches or be blocked. This helps prune VLANs that have no members on a switch. If the applicant mode is set to **participant** or **active**, you can use the [gvrp restrict-vlan-advertisement](#) command to restrict the propagation of VLAN information on a specified port as shown:

```
-> gvrp restrict-vlan-advertisement port 3/1 4
```

Here, VLAN 4 is not allowed to propagate on port 1 of slot 3.

To enable the propagation of dynamic VLANs on the specified port, use the **no** form of the command. To restrict VLAN 4 from being propagated to port 3/1, enter the command as shown:

```
-> no gvrp restrict-vlan-advertisement port 3/1 4
```

Verifying GVRP Configuration

A summary of the commands used for verifying GVRP configuration is given here:

clear gvrp statistics	Clears GVRP statistics for all the ports, an aggregate of ports, or a specific port.
show gvrp last-pdu-origin	Displays the source MAC address of the last GVRP message received on a specified port or an aggregate of ports.
show gvrp configuration	Displays the global configuration for GVRP.
show gvrp configuration port	Displays the GVRP configuration status for all the ports.
show gvrp configuration link-agg/port	Displays the GVRP configuration for a specific port or an aggregate of ports.
show gvrp timer	Displays the timer values configured for all the ports or a specific port.

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

6 Assigning Ports to VLANs

Initially all switch ports are non-mobile (fixed) and are assigned to VLAN 1, which is also their *configured default* VLAN. When additional VLANs are created on the switch, ports are assigned to the VLANs so that traffic from devices connected to these ports is bridged within the VLAN domain. Switch ports are either statically or dynamically assigned to VLANs.

Methods for statically assigning ports to VLANs include the following:

- Using the **vlan port default** command to define a new configured default VLAN for both non-mobile (fixed) and mobile ports. (See “[Statically Assigning Ports to VLANs](#)” on page 6-4.)
- Using the **vlan 802.1q** command to define tagged VLANs for non-mobile ports. This method allows the switch to bridge traffic for multiple VLANs over one physical port connection. (See [Chapter 22, “Configuring 802.1Q.”](#))
- Configuring ports as members of a link aggregate that is assigned to a configured default VLAN. (See [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#))

Dynamic assignment applies only to mobile ports. When traffic is received on a mobile port, the packets are classified using one of the following methods to determine VLAN assignment (see “[Dynamically Assigning Ports to VLANs](#)” on page 6-4 for more information):

- Packet is tagged with a VLAN ID that matches the ID of another VLAN that has mobile tagging enabled.
- Packet contents matches criteria defined in a VLAN rule.

Regardless of how a port is assigned to a VLAN, once the assignment occurs, a VLAN port association (VPA) is created and tracked by VLAN management software on each switch.

In This Chapter

This chapter describes how to statically assign ports to a new default VLAN and configure mobile ports for dynamic assignment through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Statically assigning ports to VLANs on [page 6-4](#).
- Dynamically assigning ports to VLANs (port mobility) [page 6-10](#).
- Configuring mobile port properties (including authentication) on [page 6-16](#).

Port Assignment Specifications

Note that the maximum limit values provided in the following Specifications table are subject to available system resources:

IEEE Standards Supported	802.1Q– <i>Virtual Bridged Local Area Networks</i> 802.1D– <i>Media Access Control Bridges</i>
Platforms Supported	OmniSwitch 6250 Series
Maximum VLANs per switch	4094 (based on switch configuration and available resources).
Maximum VLAN port associations (VPA) per switch	32768
Maximum 802.1Q VLAN port associations per switch	2500
Switch ports eligible for port mobility.	Untagged Ethernet and gigabit Ethernet ports that are not members of a link aggregate.
Switch ports eligible for dynamic VLAN assignment.	Mobile ports.
Switch ports eligible for static VLAN assignment.	Non-mobile (fixed) ports. Mobile ports. Uplink ports. Link aggregate of ports.

Port Assignment Defaults

Parameter Description	Command	Default
Configured default VLAN	vlan port default	All ports initially associated with default VLAN 1.
Port mobility	vlan port mobile	Disabled
Bridge mobile port traffic that doesn't match any VLAN rules on the configured default VLAN	vlan port default vlan	Disabled
Drop mobile port dynamic VLAN assignments when learned mobile port traffic that triggered the assignment ages out	vlan port default vlan restore	Enabled
Enable Layer 2 authentication on the mobile port	vlan port authenticate	Disabled
Enable 802.1x port-based access control on a mobile port	vlan port 802.1x	Disabled

Sample VLAN Port Assignment

The following steps provide a quick tutorial that will create a VLAN, statically assign ports to the VLAN, and configure mobility on some of the VLAN ports:

- 1 Create VLAN 255 with a description (e.g., Finance IP Network) using the following command:

```
-> vlan 255 name "Finance IP Network"
```

- 2 Assign switch ports 2 through 5 on slot 3 to VLAN 255 using the following command:

```
-> vlan 255 port default 3/2-5
```

VLAN 255 is now the *configured default VLAN* for ports 2 through 5 on slot 3.

- 3 Enable mobility on ports 4 and 5 on slot 3 using the following command:

```
-> vlan port mobile 3/4-5
```

- 4 Disable the default VLAN parameter for mobile ports 3/4 and 3/5 using the following command:

```
-> vlan port 3/4-5 default vlan disable
```

With this parameter disabled, VLAN 255 will not carry any traffic received on 3/4 or 3/5 that does not match any VLAN rules configured on the switch.

Note. *Optional.* To verify that ports 2 through 5 on slot 3 were assigned to VLAN 255, enter **show vlan** followed by 255 then **port**. For example:

```
-> show vlan 255 port
port      type      status
-----+-----+-----
 3/2      default   inactive
 3/3      default   inactive
 3/4      default   inactive
 3/5      default   inactive
```

To verify the mobile status of ports 4 and 5 on slot 3 and determine which mobile port parameters are enabled, enter **show vlan port mobile** followed by a slot and port number. For example:

```
-> show vlan port mobile 3/4
Mobility           : on,
Config Default Vlan: 255,
Default Vlan Enabled: off,
Default Vlan Perm  : on,
Default Vlan Restore: on,
Authentication     : off,
Ignore BPDUs       : off
```

Statically Assigning Ports to VLANs

The **vlan port default** command is used to statically assign both mobile and non-mobile ports to another VLAN. When the assignment is made, the port drops the previous VLAN assignment. For example, the following command assigns port 2 on slot 3, currently assigned to VLAN 1, to VLAN 755:

```
-> vlan 755 port default 3/2
```

Port 3/2 is now assigned to VLAN 755 and no longer associated with VLAN 1. In addition, VLAN 755 is now the new configured default VLAN for the port.

A configured default VLAN is the VLAN statically assigned to a port. Any time the **vlan port default** command is used, the VLAN assignment is static and a new configured default VLAN is defined for the port. This command is also the only way to change a non-mobile port VLAN assignment. In addition, non-mobile ports can only retain one VLAN assignment, unlike mobile ports that can dynamically associate with multiple VLANs. See [“Dynamically Assigning Ports to VLANs” on page 6-4](#) for more information about mobile ports.

Additional methods for statically assigning ports to VLANs include the following:

- Using the **vlan 802.1q** command to define tagged VLANs for non-mobile ports. This method allows the switch to bridge traffic for multiple VLANs over one physical port connection. (See [Chapter 22, “Configuring 802.1Q,”](#) for more information.)
- Configuring ports as members of a link aggregate that is assigned to a configured default VLAN. (See [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation,”](#) for more information.)

When a port is statically assigned to a VLAN, a VLAN port association (VPA) is created and tracked by VLAN management software on each switch. To display a list of all VPAs, use the **show vlan port** command. For more information, see [“Verifying VLAN Port Associations and Mobile Port Properties” on page 6-18.](#)

Dynamically Assigning Ports to VLANs

Mobile ports are the only types of ports that are eligible for dynamic VLAN assignment. When traffic received on a mobile port matches pre-defined VLAN criteria, the port and the matching traffic are assigned to the VLAN without user intervention.

By default, all switch ports are non-mobile (fixed) ports that are statically assigned to a specific VLAN and can only belong to one default VLAN at a time. The **vlan port mobile** command is used to enable mobility on a port. Once enabled, switch software classifies mobile port traffic to determine the appropriate VLAN assignment. Depending on the type of traffic classification used (VLAN rules or VLAN ID tag), mobile ports can also associate with more than one VLAN.

VLANs do not have a mobile or non-mobile distinction and there is no overall switch setting to invoke the mobile port feature. Instead, mobility is enabled on individual switch ports and rules are defined for individual VLANs to classify mobile port traffic.

When a port is dynamically assigned to a VLAN, a VLAN port association (VPA) is created and tracked by VLAN management software on each switch. To display a list of all VPAs, use the **show vlan port** command. For more information, see [“Verifying VLAN Port Associations and Mobile Port Properties” on page 6-18.](#)

How Dynamic Port Assignment Works

Traffic received on mobile ports is classified using one of the following methods:

- Packet is tagged with a VLAN ID that matches the ID of another VLAN that has mobile tagging enabled. (See [“VLAN Mobile Tag Classification” on page 6-5](#) for more information.)
- Packet contents matches criteria defined in a VLAN rule. (See [“VLAN Rule Classification” on page 6-8](#) for more information.)

Classification triggers dynamic assignment of the mobile port and qualifying traffic to the VLAN with the matching criteria. The following sections further explain the types of classification and provide examples.

VLAN Mobile Tag Classification

VLAN mobile tag classification provides a dynamic 802.1Q tagging capability. This feature allows mobile ports to receive and process 802.1Q tagged packets destined for a VLAN that has mobile tagging enabled.

The `vlan mobile-tag` command is used to enable or disable mobile tagging for a specific VLAN (see [Chapter 4, “Configuring VLANs,”](#) for more information). If 802.1Q tagging is required on a fixed (non-mobile) port, then the `vlan 802.1q` command is still used to statically tag VLANs for the port (see [Chapter 22, “Configuring 802.1Q,”](#) for more information).

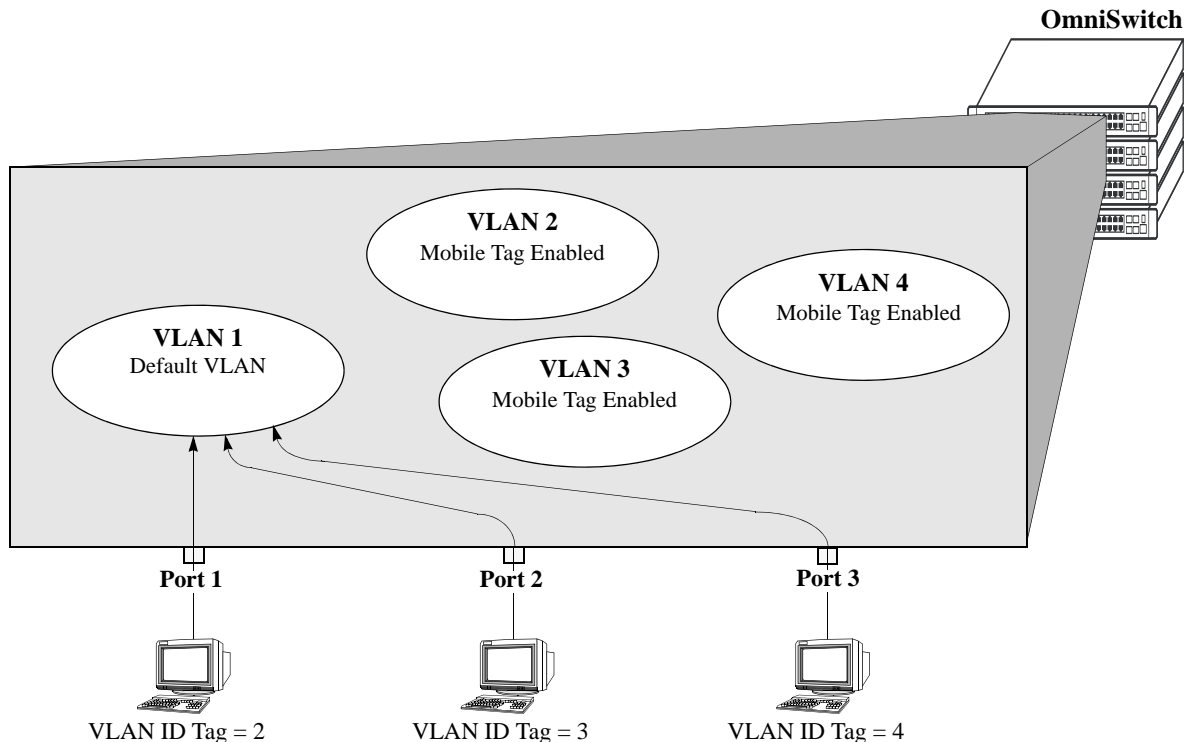
Consider the following when using VLAN mobile tag classification:

- Using mobile tagging allows the dynamic assignment of mobile ports to one or more VLANs at the same time.
- If a mobile port receives a tagged packet with a VLAN ID of a VLAN that does not have mobile tagging enabled or the VLAN does not exist, the packet is dropped.
- VLAN mobile tag classification takes precedence over VLAN rule classification. If a mobile port receives traffic that matches a VLAN rule and also has an 802.1Q VLAN ID tag for a VLAN with mobile tagging enabled, the port is dynamically assigned to the mobile tag VLAN and not the matching rule VLAN.
- If the administrative status of a mobile tag VLAN is disabled, dynamic mobile port assignments are retained but traffic on these ports is filtered for the disabled VLAN. However, the VLAN mobile tag attribute remains active and continues to classify mobile port traffic for VLAN membership.

The following example shows how mobile ports are dynamically assigned using VLAN mobile tagging to classify mobile port traffic. This example includes diagrams showing the initial VLAN port assignment configuration and a diagram showing how the configuration looks after mobile port traffic is classified.

In the initial VLAN port assignment configuration shown below,

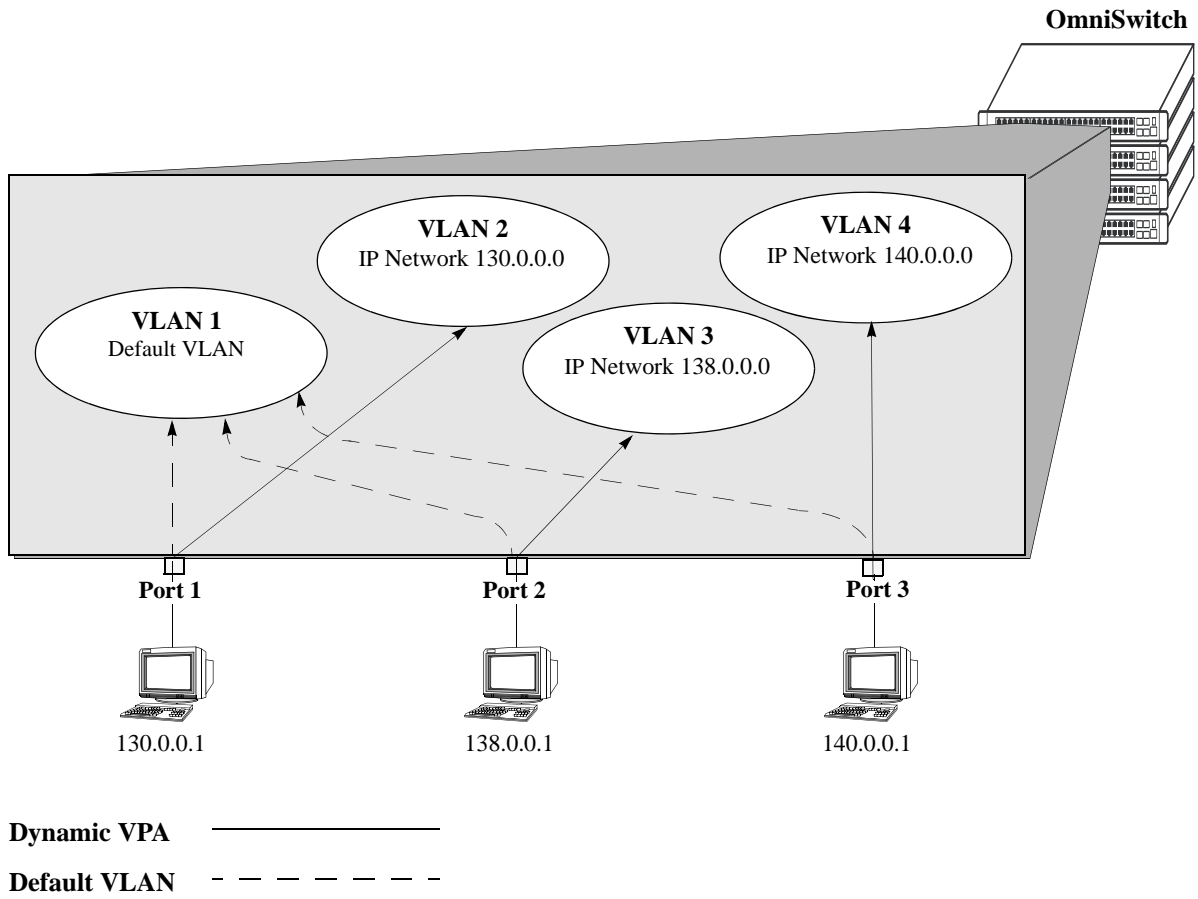
- All three ports have workstations that are configured to send packets with an 802.1Q VLAN ID tag for three different VLANs (VLAN 2, 3, and 4).
- Mobility is enabled on each of the workstation ports.
- VLAN 1 is the configured default VLAN for each port.
- VLANs 2, 3, and 4 are configured on the switch, each one has VLAN mobile tagging enabled.



VLAN Mobile Tag Classification: Initial Configuration

As soon as the workstations start sending traffic, switch software checks the 802.1Q VLAN ID tag of the frames and looks for a VLAN that has the same ID and also has mobile tagging enabled. Since the workstations are sending tagged packets destined for the mobile tag enabled VLANs, each port is assigned to the appropriate VLAN without user intervention. As the diagram on [page 6-7](#) shows,

- Port 1 is assigned to VLAN 2, because the workstation is transmitting tagged packets destined for VLAN 2.
- Port 2 is assigned to VLAN 3 because the workstation is transmitting tagged packets destined for VLAN 3.
- Port 3 is assigned to VLAN 4 because the workstation is transmitting tagged packets destined for VLAN 4.
- All three ports, however, retain their default VLAN 1 assignment, but now have an additional VLAN port assignment that carries the matching traffic on the appropriate rule VLAN.



Tagged Mobile Port Traffic Triggers Dynamic VLAN Assignment

VLAN Rule Classification

VLAN rule classification triggers dynamic VLAN port assignment when traffic received on a mobile port matches the criteria defined in a VLAN rule. Different rule types are available for classifying different types of network device traffic (see [Chapter 8, “Defining VLAN Rules,”](#) for more information).

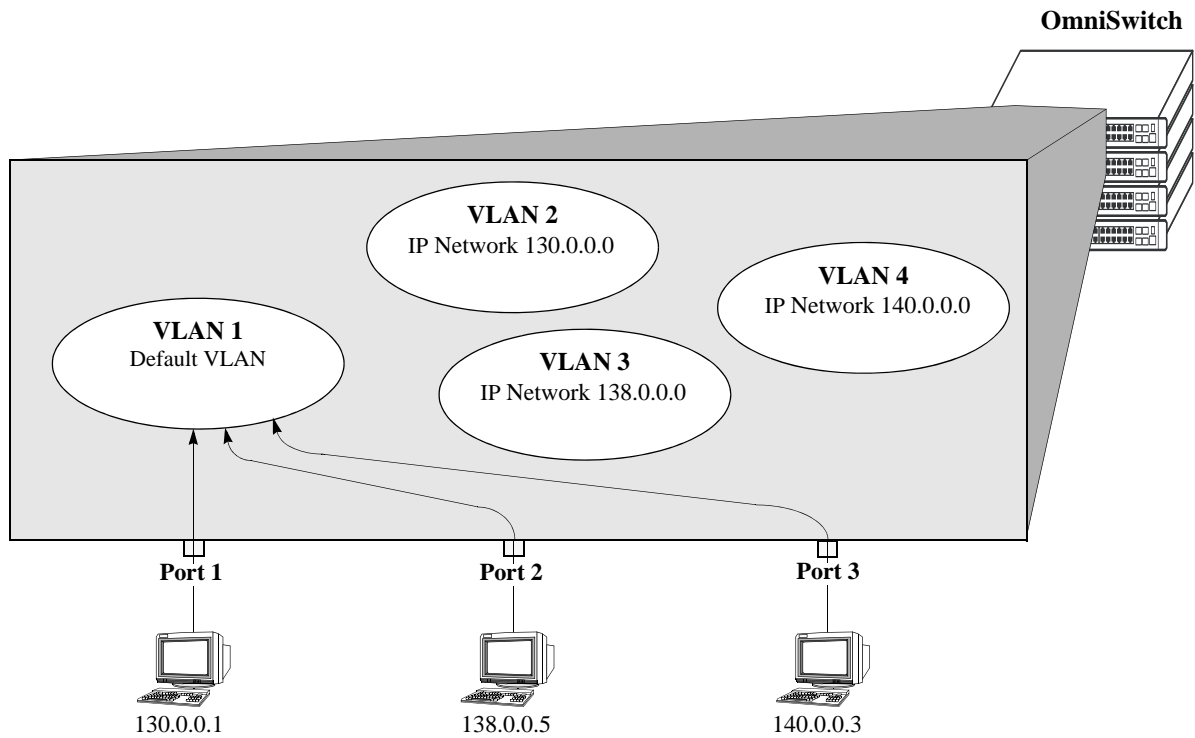
Note the following items when using VLAN rule classification:

- IP network address rules are applied to traffic received on both mobile *and* fixed ports. If traffic contains a source IP address that is included in the subnet specified by the rule, the traffic is dropped. This does not occur, however, if the IP network address rule is configured on the default VLAN for the fixed port.
- If the contents of a mobile port frame matches the values specified in both an IP network address rule and a port-protocol binding rule, the IP network address rule takes precedence. However, if the contents of such frame violates the port-protocol binding rule, the frame is dropped. See [Chapter 8, “Defining VLAN Rules,”](#) for more information about rule precedence.
- When an active device is disconnected from a mobile port and connected to a fixed port, the source MAC address of that device is not learned on the fixed port until the MAC address has aged out and no longer appears on the mobile port.
- If a VLAN is administratively disabled, dynamic mobile port assignments are retained but traffic on these ports is filtered for the disabled VLAN. However, VLAN rules remain active and continue to classify mobile port traffic for VLAN membership.
- When a VLAN is deleted from the switch configuration, all rules defined for that VLAN are automatically removed and any static or dynamic port assignments are dropped.

The following example illustrates how mobile ports are dynamically assigned using VLAN rules to classify mobile port traffic. This example includes diagrams showing the initial VLAN port assignment configuration and a diagram showing how the configuration looks after mobile port traffic is classified.

In the initial VLAN port assignment configuration shown on [page 6-9](#),

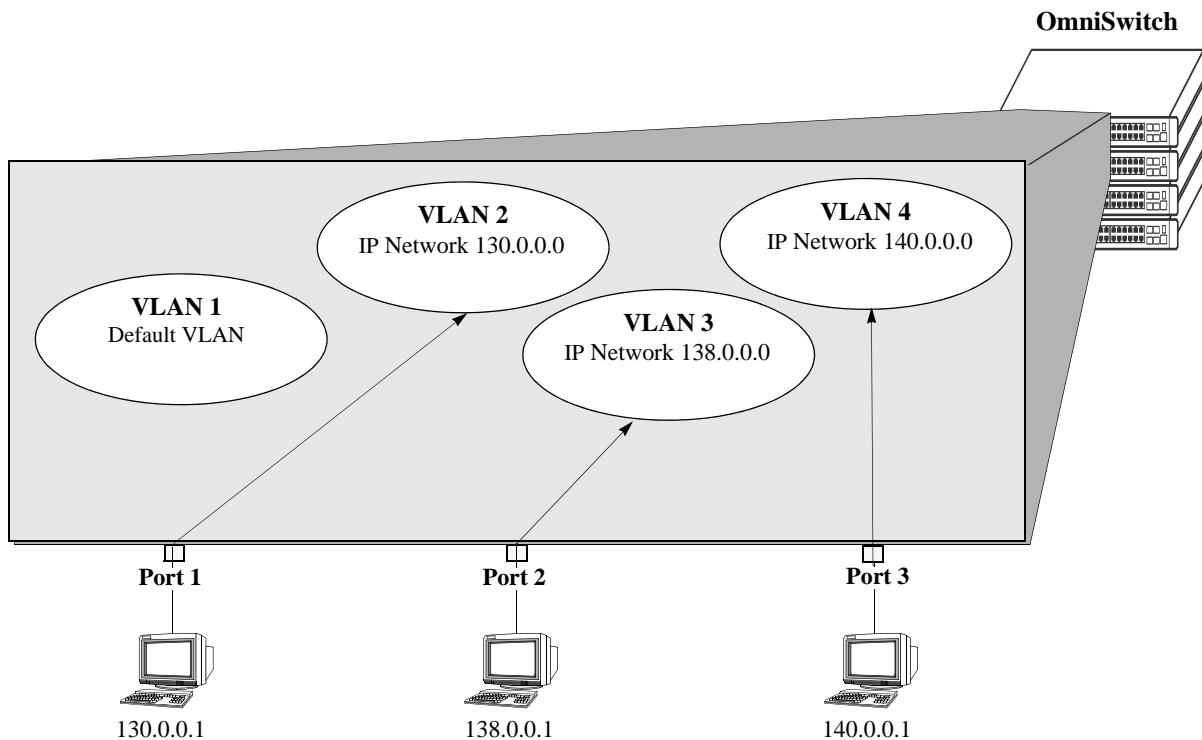
- All three ports have workstations that belong to three different IP subnets (130.0.0.0, 138.0.0.0, and 140.0.0.0).
- Mobility is enabled on each of the workstation ports.
- VLAN 1 is the configured default VLAN for each port.
- Three additional VLANs are configured on the switch, each one has an IP network address rule defined for one of the IP subnets.



VLAN Rule Classification: Initial Configuration

As soon as the workstations start sending traffic, switch software checks the source subnet of the frames and looks for a match with any configured IP network address rules. Since the workstations are sending traffic that matches a VLAN rule, each port is assigned to the appropriate VLAN without user intervention. As the diagram on [page 6-10](#) shows,

- Port 1 is assigned to VLAN 2, because the workstation is transmitting IP traffic on network 130.0.0.0 that matches the VLAN 2 network address rule.
- Port 2 is assigned to VLAN 3 because the workstation is transmitting IP traffic on network 138.0.0.0 that matches the VLAN 3 network address rule.
- Port 3 is assigned to VLAN 4 because the workstation is transmitting IP traffic on network 140.0.0.0 that matches the VLAN 4 network address rule.



Dynamic VPA —————

Default VLAN - - - - -

Mobile Port Traffic Triggers Dynamic VLAN Assignment

Configuring Dynamic VLAN Port Assignment

Dynamic VLAN port assignment requires the following configuration steps:

- 1** Use the **vlan port mobile** command to enable mobility on switch ports that will participate in dynamic VLAN assignment. See [“Enabling/Disabling Port Mobility” on page 6-11](#) for detailed procedures.
- 2** Enable/disable mobile port properties that determine mobile port behavior. See [“Configuring Mobile Port Properties” on page 6-16](#) for detailed procedures.
- 3** Create VLANs that will receive and forward mobile port traffic. See [Chapter 4, “Configuring VLANs,”](#) for more information.
- 4** Configure the method of traffic classification (VLAN rules or tagged VLAN ID) that will trigger dynamic assignment of a mobile port to the VLANs created in Step 3. See [“VLAN Rule Classification” on page 6-8](#) and [“VLAN Mobile Tag Classification” on page 6-5](#) for more information.

Once the above configuration steps are completed, dynamic VLAN assignment occurs when a device connected to a mobile port starts to send traffic. This traffic is examined by switch software to determine which VLAN should carry the traffic based on the type of classification, if any, defined for a particular VLAN. See [“Dynamically Assigning Ports to VLANs” on page 6-4](#) for more information and examples of dynamic VLAN port assignment.

Enabling/Disabling Port Mobility

To enable mobility on a port, use the **vlan port mobile** command. For example, the following command enables mobility on port 1 of slot 4:

```
-> vlan port mobile 4/1
```

To enable mobility on multiple ports, specify a range of ports and/or multiple slots.

```
-> vlan port mobile 4/1-5 5/12-20 6/10-15
```

Use the **no** form of this command to disable port mobility.

```
-> vlan no port mobile 5/21-24 6/1-4
```

Only Ethernet and gigabit Ethernet ports are eligible to become mobile ports. If any of the following conditions are true, however, these ports are considered non-mobile ports and are not available for dynamic VLAN assignment:

- The mobile status for the port is disabled (the default).
- The port is an 802.1Q tagged port.
- The port belongs to a link aggregate of ports.
- Spanning Tree is active on the port and the BPDU ignore status is disabled for the port. (See [“Ignoring Bridge Protocol Data Units \(BPDU\)” on page 6-11](#) for more information.)
- The port is configured to mirror other ports.

Note. Mobile ports are automatically *trusted* ports regardless of the QoS settings. See [Chapter 34, “Configuring QoS,”](#) for more information.

Use the **show vlan port mobile** command to display a list of ports that are mobile or are eligible to become mobile. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Ignoring Bridge Protocol Data Units (BPDU)

By default, ports that send or receive Spanning Tree Bridge Protocol Data Units (BPDU) are not eligible for dynamic VLAN assignment. If the switch sees BPDU on a port, it does not attempt to classify the port’s traffic. The **vlan port mobile** command, however, provides an optional **BPDU ignore** parameter. If this parameter is enabled when mobility is enabled on the port, the switch does not look for BPDU to determine if the port is eligible for dynamic assignment.

When **BPDU ignore** is disabled and the mobile port receives a BPDU, mobility is shut off on the port and the following occurs:

- The Switch Logging feature is notified of the port’s change in mobile status (see [Chapter 39, “Using Switch Logging,”](#) for more information).
- The port becomes a fixed (non-mobile) port that is associated only with its configured default VLAN.
- The port is included in the Spanning Tree algorithm.
- Mobility remains off on the port even if the port’s link is disabled or disconnected. Rebooting the switch, however, will restore the port’s original mobile status.

When **BPDU ignore** is enabled and the mobile port receives a BPDU, the following occurs:

- The port retains its mobile status and remains eligible for dynamic VLAN assignment.
- The port is not included in the Spanning Tree algorithm.

Note. Enabling BPDU ignore is not recommended. In specific cases where it is required, such as connecting legacy networks to mobile port networks, make sure that ignoring BPDU on a mobile port will not cause network loops to go undetected. Connectivity problems could also result if a mobile BPDU port dynamically moves out of its configured default VLAN where it provides traffic flow to/from the network.

The following command enables mobility and BPDU ignore on port 8 of slot 3:

```
-> vlan port mobile 3/8 BPDU ignore enable
```

Enabling mobility on an active port that sends or receives BPDU (e.g. ports that connect two switches and Spanning Tree is enabled on both the ports and their assigned VLANs) is not allowed. If mobility is required on this type of port, enable mobility and the **BPDU ignore** parameter when the port is not active.

Understanding Mobile Port Properties

Dynamic assignment of mobile ports occurs without user intervention when mobile port traffic matches VLAN criteria. When ports are dynamically assigned, however, the following configurable mobile port properties affect how a port uses its *configured default VLAN* and how long it retains a VLAN port association (VPA):

Mobile Port Property	If enabled	If disabled
Default VLAN	Port traffic that does not match any VLAN rules configured on the switch is flooded on the port's configured default VLAN.	Port traffic that does not match any VLAN rules is discarded.
Restore default VLAN	Port does not retain a dynamic VPA when the traffic that triggered the assignment ages out of the switch MAC address table (forwarding database).	Port retains a dynamic VPA when the qualifying traffic ages out of the switch MAC address table.

The effects of enabling or disabling mobile port properties are described through the following diagrams:

- How Mobile Port Traffic that Does Not Match any VLAN Rules is Classified on [page 6-14](#).
- How Mobile Port VLAN Assignments Age on [page 6-15](#).

What is a Configured Default VLAN?

Every switch port, mobile or non-mobile, has a configured default VLAN. Initially, this is VLAN 1 for all ports, but is configurable using the **vlan port default** command. For more information, see “[Statically Assigning Ports to VLANs](#)” on [page 6-4](#).

To view current VPA information for the switch, use the **show vlan port** command. Configured default VLAN associations are identified with a value of **default** in the **type** field. For more information, see “[Verifying VLAN Port Associations and Mobile Port Properties](#)” on [page 6-18](#).

What is a Secondary VLAN?

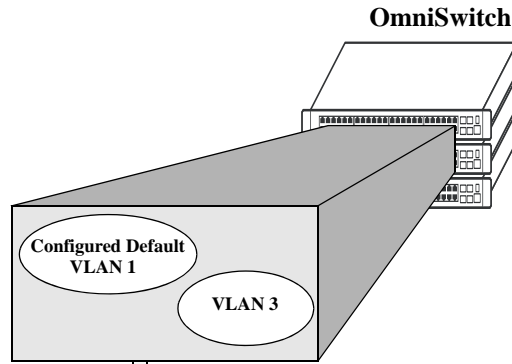
All mobile ports start out with a configured default VLAN assignment. When mobile port traffic matches VLAN criteria, the port is assigned to that VLAN. Secondary VLANs are any VLAN a port is subsequently assigned to that is not the configured default VLAN for that port.

A mobile port can obtain more than one secondary VLAN assignment under the following conditions:

- Mobile port receives untagged frames that contain information that matches rules on more than one VLAN. For example, if a mobile port receives IP and RIP frames and there is an IP protocol rule on VLAN 10 and an RIP protocol rule on VLAN 20, the mobile port is dynamically assigned to both VLANs. VLANs 10 and 20 become secondary VLAN assignments for the mobile port.
- Mobile port receives 802.1Q tagged frames that contain a VLAN ID that matches a VLAN that has VLAN mobile tagging enabled. For example, if a mobile port receives frames tagged for VLAN 10, 20 and 30 and these VLANs have mobile tagging enabled, the mobile port is dynamically assigned to all three VLANs. VLANs 10, 20, and 30 become secondary VLAN assignments for the mobile port.

VLAN Management software on each switch tracks VPAs. When a mobile port link is disabled and then enabled, all secondary VLAN assignments for that port are automatically dropped and the port's original configured default VLAN assignment is restored. Switch ports are disabled when a device is disconnected from the port, a configuration change is made to disable the port, or switch power is turned off.

To view current VPA information for the switch, use the [show vlan port](#) command. Dynamic secondary VLAN associations are identified with a value of **mobile** in the **type** field. For more information, see [“Verifying VLAN Port Associations and Mobile Port Properties”](#) on page 6-18.

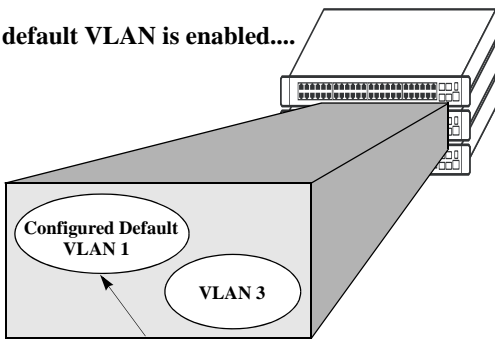


Device connected to a mobile port sends traffic. If the traffic matches existing VLAN criteria, then the mobile port and its traffic are dynamically assigned to that VLAN.



If device traffic does not match any VLAN rules, then the default VLAN property determines if the traffic is forwarded on the port's configured default VLAN (VLAN 1 in this example).

If default VLAN is enabled....



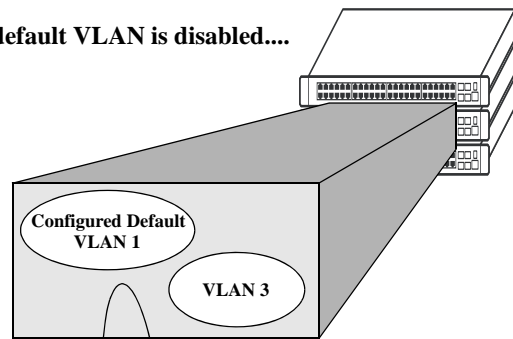
Device traffic that does not match any VLAN rules is forwarded on the mobile port's configured default VLAN.



Why enable default VLAN?

Ensures that all mobile port device traffic is carried on at least one VLAN.

If default VLAN is disabled....



Device traffic that does not match any VLAN rules is discarded.

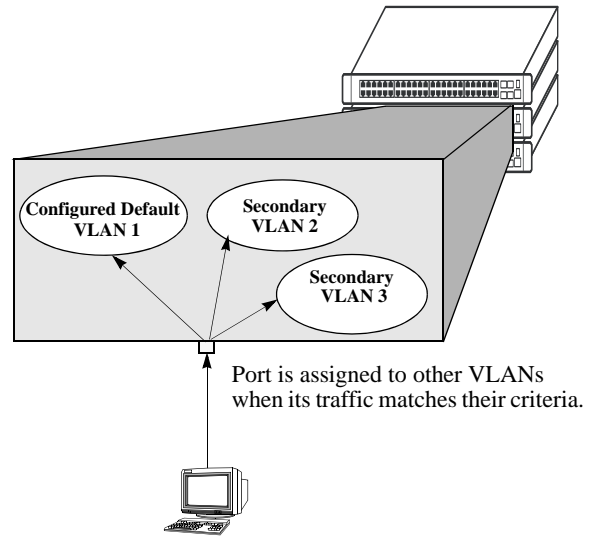
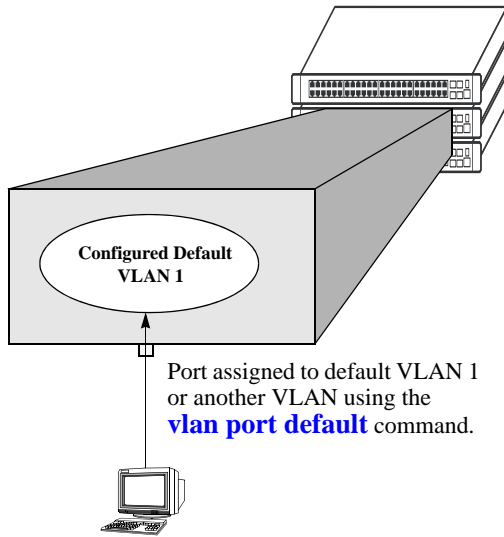


Why disable default VLAN?

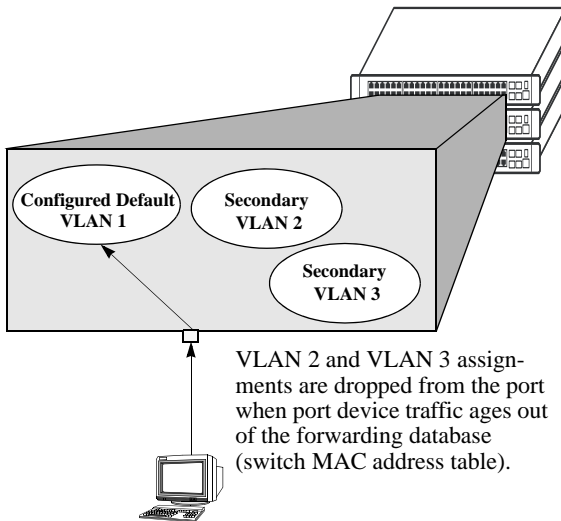
Reduces unnecessary traffic flow on a port's configured default VLAN.

Restricts dynamic assignment to mobile port traffic that matches one or more VLAN rules.

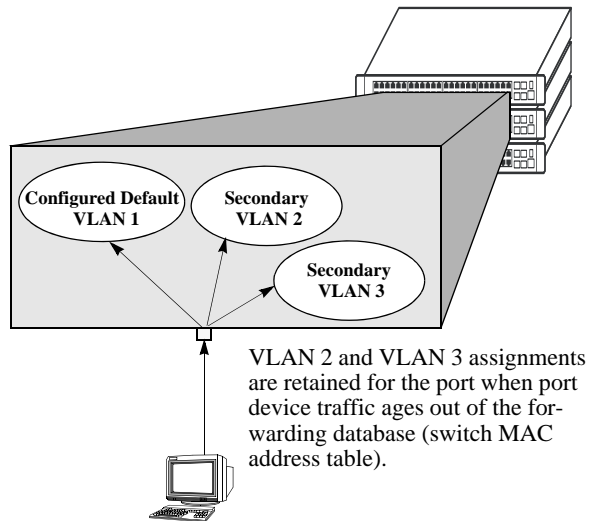
How Mobile Port Traffic that Does Not Match any VLAN Rules is Classified



If restore default VLAN is enabled....



If restore default VLAN is disabled....



Why enable restore default VLAN?

Security. VLANs only contain mobile port traffic that has recently matched rule criteria.

VPAs created from occasional network users (e.g., laptop) are not unnecessarily retained.

Why disable restore default VLAN?

VPAs are retained even when port traffic is idle for some time. When traffic resumes, it is not necessary to relearn the same VPA again. Appropriate for devices that only send occasional traffic.

How Mobile Port VLAN Assignments Age

Configuring Mobile Port Properties

Mobile port properties indicate mobile port status and affect port behavior when the port is dynamically assigned to one or more VLANs. For example, mobile port properties determine the following:

- Should the configured default VLAN forward or discard port traffic that does not match any VLAN rule criteria.
- Should the port retain or drop a dynamic VPA when traffic that triggered the assignment stops and the source MAC address learned on the port for that VLAN is aged out. (See [Chapter 2, “Managing Source Learning,”](#) for more information about the aging of MAC addresses.)

This section contains procedures for using the following commands to configure mobile port properties. For more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Command	Description
<code>vlan port default vlan</code>	Enables or disables forwarding of mobile port traffic on the port's configured default VLAN that does not match any existing VLAN rules.
<code>vlan port default vlan restore</code>	Enables or disables the retention of VLAN port assignments when mobile port traffic ages out.
<code>vlan port authenticate</code>	Enables or disables authentication on a mobile port.
<code>vlan port 802.1x</code>	Enables or disables 802.1X port-based access control on a mobile port.

Use the `show vlan port mobile` command to view the current status of these properties for one or more mobile ports. See [“Verifying VLAN Port Associations and Mobile Port Properties”](#) on page 6-18 for more information.

Enable/Disable Default VLAN

To enable or disable forwarding of mobile port traffic that does not match any VLAN rules on the port's configured default VLAN, enter `vlan port` followed by the port's `slot/port` designation then `default vlan` followed by `enable` or `disable`. For example,

```
-> vlan port 3/1 default vlan enable
-> vlan port 5/2 default vlan disable
```

To enable or disable the configured default VLAN on multiple ports, specify a range of ports and/or multiple slots.

```
-> vlan port 2/1-12 3/10-24 4/3-14 default vlan enable
```

Note. It is recommended that mobile ports with their default VLAN disabled should not share a VLAN with any other types of ports (e.g., mobile ports with default VLAN enabled or non-mobile, fixed ports).

See [“Understanding Mobile Port Properties”](#) on page 6-12 for an overview and illustrations of how this property affects mobile port behavior.

Enable/Disable Default VLAN Restore

To enable or disable default VLAN restore, enter **vlan port** followed by the port's **slot/port** designation then **default vlan restore** followed by **enable** or **disable**. For example,

```
-> vlan port 3/1 default vlan restore enable
-> vlan port 5/2 default vlan restore disable
```

To enable or disable default VLAN restore on multiple ports, specify a range of ports and/or multiple slots.

```
-> vlan port 2/1-12 3/10-24 4/3-14 default vlan restore enable
```

Note the following when changing the restore default VLAN status for a mobile port:

- If a hub is connected to a mobile port, enabling default VLAN restore on that port is recommended.
- VLAN port rule assignments are exempt from the effects of the restore default VLAN status. See [Chapter 8, “Defining VLAN Rules,”](#) for more information about using port rules to forward mobile port traffic.
- When a mobile port link is disabled and then enabled, all secondary VPAs for that port are automatically dropped regardless of the restore default VLAN status for that port. Switch ports are disabled when a device is disconnected from the port, a configuration change is made to disable the port, or switch power is turned off.

See [“Understanding Mobile Port Properties”](#) on page 6-12 for an overview and illustrations of how this property affects mobile port behavior.

Enable/Disable 802.1X Port-Based Access Control

To enable or disable 802.1X on a mobile port, enter **vlan port** followed by the port's **slot/port** designation then **802.1x** followed by **enable** or **disable**. For example,

```
-> vlan port 3/1 802.1x enable
-> vlan port 5/2 802.1x disable
```

To enable or disable 802.1X on multiple ports, specify a range of ports and/or multiple slots.

```
-> vlan port 6/1-32 8/10-24 9/3-14 802.1x enable
-> vlan port 5/3-6 9/1-4 802.1x disable
```

Only mobile ports are eligible for 802.1X port-based access control. If enabled, the mobile port participates in the authentication and authorization process defined in the IEEE 802.1X standard and supported by Alcatel-Lucent switches. For more information, see [Chapter 32, “Configuring 802.1X.”](#)

Verifying VLAN Port Associations and Mobile Port Properties

To display a list of VLAN port assignments or the status of mobile port properties, use the show commands listed below:

show vlan port	Displays a list of VLAN port assignments, including the type and status for each assignment.
show vlan port mobile	Displays the mobile status and current mobile parameter values for each port.

Understanding 'show vlan port' Output

Each line of the **show vlan port** command display corresponds to a single VLAN port association (VPA). In addition to showing the VLAN ID and slot/port number, the VPA type and current status of each association are also provided.

The VPA type indicates that one of the following methods was used to create the VPA:

Type	Description
default	The port was statically assigned to the VLAN using the vlan port default command. The VLAN is now the port's configured default VLAN.
qtagged	The port was statically assigned to the VLAN using the vlan 802.1q command. The VLAN is a static secondary VLAN for the 802.1Q tagged port.
mobile	The port is mobile and was dynamically assigned when traffic received on the port matched VLAN criteria (VLAN rules or tagged VLAN ID). The VLAN is a dynamic secondary VLAN assignment for the mobile port.
mirror	The port is assigned to the VLAN because it is configured to mirror another port that is assigned to the same VLAN. For more information about the Port Mirroring feature, see Chapter 38, "Diagnosing Switch Problems."

The VPA status indicates one of the following:

Status	Description
inactive	Port is not active (administratively disabled, down, or nothing connected to the port) for the VPA.
blocking	Port is active, but not forwarding traffic for the VPA.
forwarding	Port is forwarding all traffic for the VPA.
filtering	Mobile port traffic is filtered for the VPA; only traffic received on the port that matches VLAN rules is forwarded. Occurs when a mobile port's VLAN is administratively disabled or the port's default VLAN status is disabled. Does not apply to fixed ports.

The following example uses the **show vlan port** command to display VPA information for all ports in VLAN 200:

```
-> show vlan 200 port

  port      type      status
-----+-----+-----
   3/24    default    inactive
   5/11    mobile     forwarding
   5/12    qtagged    blocking
```

The above example output provides the following information:

- VLAN 200 is the configured default VLAN for port 3/24, which is currently not active.
- VLAN 200 is a secondary VLAN for mobile port 5/11, which is currently forwarding traffic for this VPA.
- VLAN 200 is an 802.1Q tagged VLAN for port 5/12, which is an active port but currently blocked from forwarding traffic.

Another example of the output for the **show vlan port** command is also given in [“Sample VLAN Port Assignment” on page 6-3](#). For more information about the resulting display from this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Understanding ‘show vlan port mobile’ Output

The **show vlan port mobile** command provides information regarding a port’s mobile status. If the port is mobile, the resulting display also provides the current status of the port’s mobile properties. The following example displays mobile port status and property values for ports 8/2 through 8/5:

```
-> show vlan port mobile

      cfg                ignore
  port  mobile  def  authent  enabled  restore  bpdu
-----+-----+-----+-----+-----+-----+-----
   8/2   on    200   off     off     on       off
   8/3   on    200   off     on      off     off
   8/4   on    200 on-8021x  on      off     off
```

Note that the **show vlan port mobile** command only displays ports that are mobile or are eligible to become mobile ports. For example, ports that are part of a link aggregate or are configured for 802.1Q VLAN tagging are not included in the output of this command.

Another example of the output for the **show vlan port mobile** command is also given in [“Sample VLAN Port Assignment” on page 6-3](#). For more information about the resulting display from this command, see the *OmniSwitch 6250 CLI Reference Guide*.

7 Configuring Port Mapping

Port Mapping is a security feature, which controls communication between peer users. Each session comprises a session ID, a set of user ports, and/or a set of network ports. The user ports within a session cannot communicate with each other and can only communicate via network ports. In a port mapping session with user port set A and network port set B, the ports in set A can only communicate with the ports in set B. If set B is empty, the ports in set A can communicate with rest of the ports in the system.

A port mapping session can be configured in the unidirectional or bidirectional mode. In the unidirectional mode, the network ports can communicate with each other within the session. In the bidirectional mode, the network ports cannot communicate with each other. Network ports of a unidirectional port mapping session can be shared with other unidirectional sessions, but cannot be shared with any sessions configured in the bidirectional mode. Network ports of different sessions can communicate with each other.

In This Chapter

This chapter describes the port mapping security feature and explains how to configure the same through the Command Line Interface (CLI).

Configuration procedures described in this chapter include:

- [Creating/Deleting a Port Mapping Session](#)—see [“Creating a Port Mapping Session”](#) on page 7-3 or [“Deleting a Port Mapping Session”](#) on page 7-3.
- [Enabling/Disabling a Port Mapping Session](#)—see [“Enabling a Port Mapping Session”](#) on page 7-4 or [“Disabling a Port Mapping Session”](#) on page 7-4.
- [Configuring a Port Mapping Direction](#)—see [“Configuring Unidirectional Port Mapping”](#) on page 7-4 and [“Restoring Bidirectional Port Mapping”](#) on page 7-4.
- [Configuring an example Port Mapping Session](#)—see [“Sample Port Mapping Configuration”](#) on page 7-5.
- [Verifying a Port Mapping Session](#)—see [“Verifying the Port Mapping Configuration”](#) on page 7-6.

Port Mapping Specifications

Platforms Supported	OmniSwitch 6250 Series
Ports Supported	Ethernet (10 Mbps)/Fast Ethernet (100 Mbps)/Gigabit Ethernet (1 Gb/1000 Mbps)
Mapping Sessions	Eight sessions supported per standalone switch and stack.

Port Mapping Defaults

The following table shows port mapping default values.

Parameter Description	CLI Command	Default Value/Comments
Mapping Session Creation	<code>port mapping user-port network-port</code>	No mapping sessions
Mapping Status configuration	<code>port mapping</code>	Disabled
Port Mapping Direction	<code>port mapping</code>	Bidirectional

Quick Steps for Configuring Port Mapping

Follow the steps below for a quick tutorial on configuring port mapping sessions. Additional information on how to configure each command is given in the subsections that follow.

- 1 Create a port mapping session with/without, user/network ports with the `port mapping user-port network-port` command. For example:

```
-> port mapping 8 user-port 1/2 network-port 1/3
```

- 2 Enable the port mapping session with the `port mapping` command. For example:

```
-> port mapping 8 enable
```

Note. You can verify the configuration of the port mapping session by entering `show port mapping` followed by the session ID.

```
-> show port mapping 3
```

```

SessionID      USR-PORT      NETWORK-PORT
-----+-----+-----
      8          1/2          1/3

```

You can also verify the status of a port mapping session by using the `show port mapping status` command.

Creating/Deleting a Port Mapping Session

Before port mapping can be used, it is necessary to create a port mapping session. The following subsections describe how to create and delete a port mapping session with the **port mapping user-port network-port** and **port mapping** command, respectively.

Creating a Port Mapping Session

To create a port mapping session either with or without the user ports, network ports, or both, use the **port mapping user-port network-port** command. For example, to create a port mapping session 8 with a user port on slot 1 port 2 and a network port on slot 1 port 3, you would enter:

```
-> port mapping 8 user-port 1/2 network-port 1/3
```

You can create a port mapping session with link aggregate network ports. For example, to create a port mapping session 3 with network ports of link aggregation group 7, you would enter:

```
-> port mapping 3 network-port linkagg 7
```

You can specify all the ports of a slot to be assigned to a mapping session. For example, to create a port mapping session 3 with all the ports of slot 1 as network ports, you would enter:

```
-> port mapping 3 network-port slot 1
```

You can specify a range of ports to be assigned to a mapping session. For example, to create a port mapping session 4 with ports 5 through 8 on slot 2 as user ports, you would enter:

```
-> port mapping 4 user-port 2/5-8
```

Deleting a User/Network Port of a Session

To delete a user/network port of a port mapping session, use the **no** form of the **port mapping user-port network-port** command. For example, to delete a user port on slot 1 port 3 of a mapping session 8, you would enter:

```
-> port mapping 8 no user-port 1/3
```

Similarly, to delete the network ports of link aggregation group 7 of a mapping session 4, you would enter:

```
-> port mapping 4 no network-port linkagg 7
```

Deleting a Port Mapping Session

To delete a previously created mapping session, use the **no** form of the **port mapping** command. For example, to delete the port mapping session 6, you would enter:

```
-> no port mapping 6
```

Note. You must delete any attached ports with the **port mapping user-port network-port** command before you can delete a port mapping session.

Enabling/Disabling a Port Mapping Session

By default, the port mapping session will be disabled. The following subsections describe how to enable and disable the port mapping session with the **port mapping** command.

Enabling a Port Mapping Session

To enable a port mapping session, enter **port mapping** followed by the session ID and **enable**. For example, to enable the port mapping session 5, you would enter:

```
-> port mapping 5 enable
```

Disabling a Port Mapping Session

To disable a port mapping session, enter **port mapping** followed by the session ID and **disable**. For example, to disable the port mapping session 5, you would enter:

```
-> port mapping 5 disable
```

Configuring a Port Mapping Direction

By default, port mapping sessions are bidirectional. The following subsections describe how to configure and restore the directional mode of a port mapping session with the **port mapping** command.

Configuring Unidirectional Port Mapping

To configure a unidirectional port mapping session, enter **port mapping** followed by the session ID and **unidirectional**. For example, to configure the direction of a port mapping session 6 as unidirectional, you would enter:

```
-> port mapping 6 unidirectional
```

Restoring Bidirectional Port Mapping

To restore the direction of a port mapping session to its default (i.e., bidirectional), enter **port mapping** followed by the session ID and **bidirectional**. For example, to restore the direction (i.e., bidirectional) of the port mapping session 5, you would enter:

```
-> port mapping 5 bidirectional
```

Note. To change the direction of an active session with network ports, delete the network ports of the session, change the direction, and recreate the network ports.

Sample Port Mapping Configuration

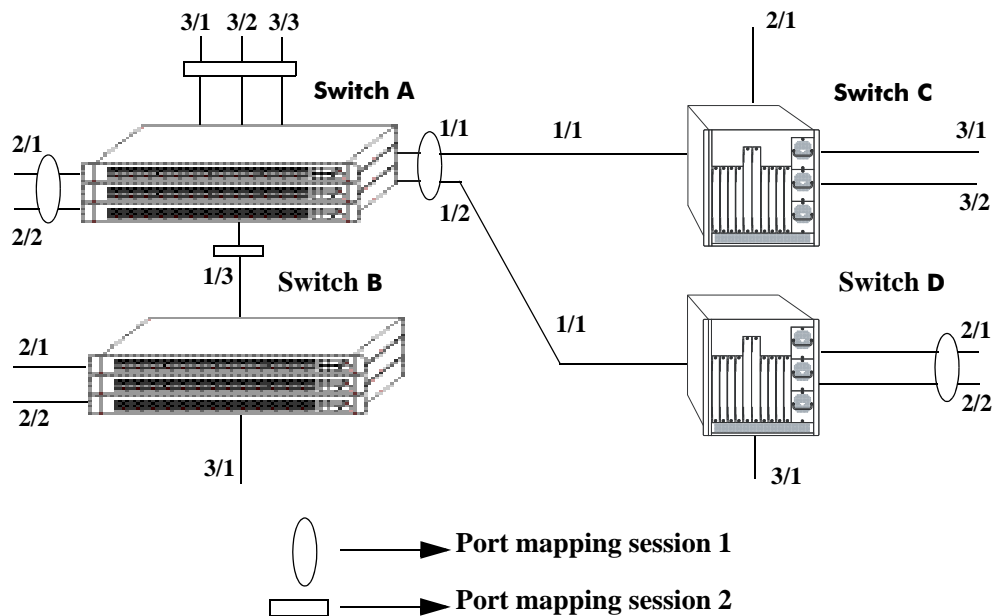
This section provides an example port mapping network configuration. In addition, a tutorial is also included that provides steps on how to configure the example port mapping session using the Command Line Interface (CLI).

Example Port Mapping Overview

The following diagram shows a four-switch network configuration with active port mapping sessions. In the network diagram, the Switch A is configured as follows:

- Port mapping session 1 is created with user ports 2/1, 2/2 and network ports 1/1, 1/2 and is configured in the unidirectional mode.
- Port mapping session 2 is created with user ports 3/1, 3/2, and 3/3 and network port 1/3.

The Switch D is configured by creating a port mapping session 1 with user ports 2/1, 2/2 and network ports 1/1.



Example Port Mapping Topology

In the above example topology:

- Ports 2/1 and 2/2 on Switch A do not interact with each other and do not interact with the ports on Switch B.
- Ports 2/1, 2/2, and 3/1 on Switch B interact with all the ports of the network except with ports 2/1 and 2/2 on Switch A.
- Ports 2/1 and 2/2 on Switch D do not interact with each other but they interact with all the user ports on Switch A except 3/1, 3/2, and 3/3. They also interact with all the ports on Switch B and Switch C.
- Ports 3/1, 3/2, and 2/1 on Switch C can interact with all the user ports on the network except 3/1, 3/2, and 3/3 on Switch A.

Example Port Mapping Configuration Steps

The following steps provide a quick tutorial that configures the port mapping session shown in the diagram on [page 7-5](#).

- 1 Configure session 1 on Switch A in the unidirectional mode using the following command:

```
-> port mapping 1 unidirectional
```

- 2 Create two port mapping sessions on Switch A using the following commands:

```
-> port mapping 1 user-port 2/1-2 network-port 1/1-2
```

```
-> port mapping 2 user-port 3/1-3 network-port 1/3
```

- 3 Enable both the sessions on Switch A using the following commands:

```
-> port mapping 1 enable
```

```
-> port mapping 2 enable
```

Similarly, create and enable a port mapping session 1 on Switch D by entering the following commands:

```
-> port mapping 1 user-port 2/1-2 network-port 1/1
```

```
-> port mapping 1 enable
```

Verifying the Port Mapping Configuration

To display information about the port mapping configuration on the switch, use the show commands listed below:

- | | |
|---------------------------------|------------------------------------------------------------------|
| show port mapping status | Displays the status of one or more port mapping sessions. |
| show port mapping | Displays the configuration of one or more port mapping sessions. |

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

8 Defining VLAN Rules

VLAN rules are used to classify mobile port traffic for dynamic VLAN port assignment. Rules are defined by specifying a port, MAC address, protocol, network address, or DHCP criteria to capture certain types of network device traffic. It is also possible to define multiple rules for the same VLAN. A mobile port is assigned to a VLAN if its traffic matches any one VLAN rule.

There is an additional method for dynamically assigning mobile ports to VLANs that involves enabling VLAN mobile tagging. This method is similar to defining rules in that the feature is enabled on the VLAN that is going to receive the mobile port tagged traffic. The difference, however, is that tagged packets received on mobile ports are classified by their 802.1Q VLAN ID tag and not by whether or not their source MAC, network address, or protocol type matches VLAN rule criteria.

In This Chapter

This chapter contains information and procedures for defining VLAN rules through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*. Refer to [Chapter 4, “Configuring VLANs,”](#) and [Chapter 6, “Assigning Ports to VLANs,”](#) for information about the VLAN mobile tagging feature.

Configuration procedures described in this chapter include:

- Defining DHCP rules on [page 8-9](#).
- Defining MAC address rules on [page 8-10](#).
- Defining IP network address rules on [page 8-11](#).
- Defining protocol rules on [page 8-12](#).
- Defining forwarding-only port rules on [page 8-13](#).
- Verifying the VLAN rule configuration on [page 8-17](#).

For information about creating and managing VLANs, see [Chapter 4, “Configuring VLANs.”](#)

For information about enabling port mobility and defining mobile port properties, see [Chapter 6, “Assigning Ports to VLANs.”](#)

VLAN Rules Specifications

Note that the maximum limit values provided in the following Specifications table are subject to available system resources.

IEEE Standards Supported	802.1Q– <i>Virtual Bridged Local Area Networks</i> 802.1v– <i>VLAN Classification by Protocol and Port</i> 802.1D– <i>Media Access Control Bridges</i>
Platforms Supported	OmniSwitch 6250 Series
Maximum number of VLANs per switch	4094 (based on switch configuration and available resources)
Maximum number of rules per VLAN	Unlimited
Maximum number of rules per switch	8129 of each rule type with the following exceptions: <ul style="list-style-type: none"> • 1 DHCP generic rule (only one is needed) • 256 MAC and IP rules • 8 port-protocol rules
Switch ports that are eligible for VLAN rule classification (dynamic VLAN assignment)	Mobile 10/100 Ethernet and gigabit ports.
Switch ports that are not eligible for VLAN rule classification	Non-mobile (fixed) ports. Uplink/stack ports. 802.1Q tagged fixed ports. Link aggregate ports.
CLI Command Prefix Recognition	All VLAN management commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

VLAN Rules Defaults

Parameter Description	Command	Default
IP network address rule subnet mask	vlan ip	The IP address class range; Class A, B, or C.

Sample VLAN Rule Configuration

The following steps provide a quick tutorial that will create an IP network address and DHCP MAC range rule for VLAN 255. The remaining sections of this chapter provide further explanation of all VLAN rules and how they are defined.

- 1 Create VLAN 255 with a description (e.g., Finance IP Network) using the following command:

```
-> vlan 255 name "Finance IP Network"
```

- 2 Define an IP network address rule for VLAN 255 that will capture mobile port traffic containing a network 21.0.0.0 IP source address. For example:

```
-> vlan 255 ip 21.0.0.0
```

- 3 Define a DHCP MAC range rule for VLAN 255 that will capture mobile port DHCP traffic that contains a source MAC address that falls within the range specified by the rule. For example:

```
-> vlan 255 dhcp mac 00:DA:95:00:59:10 00:DA:95:00:59:9F
```

Note-Optional. To verify that the rules in this tutorial were defined for VLANs 255, 355, and 1500, enter **show vlan rules**. For example:

```
-> show vlan rules
```

type	vlan	rule
ip-net	255	21.0.0.0, 255.0.0.0
dhcp-mac-range	255	00:da:95:00:59:10, 00:da:95:00:59:9f

VLAN Rules Overview

The mobile port feature available on the switch allows dynamic VLAN port assignment based on VLAN rules that are applied to mobile port traffic. When a port is defined as a mobile port, switch software compares traffic coming in on that port with configured VLAN rules. If any of the mobile port traffic matches any of the VLAN rules, the port and the matching traffic become a member of that VLAN.

VLANs do not have a mobile or non-mobile distinction and there is no overall switch setting to invoke the mobile port feature. Instead, mobility is enabled on individual switch ports and rules are defined for individual VLANs to capture mobile port traffic. Refer to [Chapter 6, “Assigning Ports to VLANs,”](#) for more information about using mobile ports and dynamic VLAN port assignments.

VLAN Rule Types

There are several types of configurable VLAN rules available for classifying different types of network device traffic. There is no limit to the number of rules allowed per VLAN and up to 8,129 of each rule type is allowed per switch. See [“Configuring VLAN Rule Definitions” on page 8-8](#) for instructions on how to create a VLAN rule.

The type of rule defined determines the type of traffic that will trigger a dynamic port assignment to the VLAN and the type of traffic the VLAN will forward within its domain. Refer to the following sections (listed in the order of rule precedence) for a description of each type of VLAN rule:

Rule	See
DHCP MAC Address DHCP MAC Range DHCP Port DHCP Generic	“DHCP Rules” on page 8-5
MAC Address MAC Address Range	“MAC Address Rules” on page 8-5
Network Address	“Network Address Rules” on page 8-5
Protocol	“Protocol Rules” on page 8-5
Port	“Port Rules” on page 8-6

Use the [show vlan rules](#) command to display a list of rules already configured on the switch. For more information about this command, refer to the *OmniSwitch 6250 CLI Reference Guide*.

DHCP Rules

Dynamic Host Configuration Protocol (DHCP) frames are sent from client workstations to request an IP address from a DHCP server. The server responds with the same type of frames, which contain an IP address for the client. If clients are connected to mobile ports, DHCP rules are used to classify this type of traffic for the purposes of transmitting and receiving DHCP frames to and from the server.

When a mobile port receives a DHCP frame that matches a DHCP rule, the port is temporarily assigned to the VLAN long enough to forward the DHCP requests within the VLAN broadcast domain. The source MAC address of the DHCP frame, however, is not learned for that VLAN port association. As a result, the [show mac-address-table](#) command output will not contain an entry for the DHCP source MAC address. The [show vlan port](#) command output, however, will contain an entry for the temporary VLAN port association that occurs during this process.

Once a device connected to a mobile port receives an IP address from the DHCP server, the VLAN port assignment triggered by the device's DHCP frames matching a VLAN DHCP rule is dropped unless regular port traffic matches another rule on that same VLAN. If this match occurs, or the traffic matches a rule on another VLAN, then the source MAC address of the mobile port's frames is learned for that VLAN port association.

DHCP rules are most often used in combination with IP network address rules. A DHCP client has an IP address of all zeros (0.0.0.0) until it receives an IP address from a DHCP server, so initially it would not match any IP network address rules.

MAC address rules, and protocol rules also capture DHCP client traffic. The following DHCP rule types are available:

- DHCP MAC Address
- DHCP MAC Range
- DHCP Port
- DHCP Generic

MAC Address Rules

MAC address rules determine VLAN assignment based on a device's source MAC address. This is the simplest type of rule and provides the maximum degree of control and security. Members of the VLAN will consist of devices with specific MAC addresses. In addition, once a device joins a MAC address rule VLAN, it is not eligible to join multiple VLANs even if device traffic matches other VLAN rules.

MAC address rules also capture DHCP traffic, if no other DHCP rule exists that would classify the DHCP traffic into another VLAN. Therefore, it is not necessary to combine DHCP rules with MAC address rules for the same VLAN.

Network Address Rules

An IP network address rule determines VLAN mobile port assignment based on a device's source IP address.

Protocol Rules

Protocol rules determine VLAN assignment based on the protocol a device uses to communicate. When defining this type of rule, there are several generic protocol values to select from: IP, AppleTalk, or DECNet. If none of these are sufficient, it is possible to specify an Ethernet type, Destination and Source Service Access Protocol (DSAP/SSAP) header values, or a Sub-network Access Protocol (SNAP) type.

Note that specifying a SNAP protocol type restricts classification of mobile port traffic to the ethertype value found in the IEEE 802.2 SNAP LLC frame header.

IP protocol rules also capture DHCP traffic, if no other DHCP rule exists that would classify the DHCP traffic into another VLAN. Therefore, it is not necessary to combine DHCP rules with IP protocol rules for the same VLAN.

Port Rules

Port rules are fundamentally different from all other supported rule types, in that traffic is not required to trigger dynamic assignment of the mobile port to a VLAN. As soon as this type of rule is created, the specified port is assigned to the VLAN only for the purpose of forwarding broadcast types of VLAN traffic to a device connected to that same port.

Port rules are mostly used for silent devices, such as printers, that require VLAN membership to receive traffic forwarded from the VLAN. These devices usually don't send traffic, so they do not trigger dynamic assignment of their mobile ports to a VLAN.

It is also possible to specify the same port in more than one port rule defined for different VLANs. The advantage to this is that traffic from multiple VLANs is forwarded out the one mobile port to the silent device. For example, if port 3 on slot 2 is specified in a port rule defined for VLANs 255, 355, and 755, then outgoing traffic from all three of these VLANs is forwarded on port 2/3.

Port rules only apply to outgoing mobile port traffic and do not classify incoming traffic. If a mobile port is specified in a port rule, its incoming traffic is still classified for VLAN assignment in the same manner as all other mobile port traffic.

VLAN assignments that are defined using port rules are exempt from the port's default VLAN restore status. See [Chapter 6, "Assigning Ports to VLANs,"](#) for more information regarding a port's default VLAN restore status and other mobile port properties.

Understanding VLAN Rule Precedence

In addition to configurable VLAN rule types, there are two internal rule types for processing mobile port frames. One is referred to as *frame type* and is used to identify Dynamic Host Configuration Protocol (DHCP) frames. The second internal rule is referred to as *default* and identifies frames that do not match any VLAN rules.

Note. Another type of mobile traffic classification, referred to as VLAN mobile tagging, takes precedence over all VLAN rules. If a mobile port receives an 802.1Q packet that contains a VLAN ID tag that matches a VLAN that has mobile tagging enabled, the port and its traffic are assigned to this VLAN, even if the traffic matches a rule defined on any other VLAN. See [Chapter 6, "Assigning Ports to VLANs,"](#) for more information about VLAN mobile tag classification.

The VLAN rule precedence table on [page 8-7](#) provides a list of all VLAN rules, including the two internal rules mentioned above, in the order of precedence that switch software applies to classify mobile port frames. The first column lists the rule type names, the second and third columns describe how the switch handles frames that match or don't match rule criteria. The higher the rule is in the list, the higher its level of precedence.

When a frame is received on a mobile port, switch software starts with rule one in the rule precedence table and progresses down the list until there is a successful match between rule criteria and frame contents.

Precedence Step/Rule Type	Condition	Result
1. Frame Type	Frame is a DHCP frame.	Go to Step 2.
	Frame is not a DHCP frame.	Skip Steps 2, 3, 4, and 5.
2. DHCP MAC	DHCP frame contains a matching source MAC address.	Frame source is assigned to the rule's VLAN, but not learned.
3. DHCP MAC Range	DHCP frame contains a source MAC address that falls within a specified range of MAC addresses.	Frame source is assigned to the rule's VLAN, but not learned.
4. DHCP Port	DHCP frame matches the port specified in the rule.	Frame source is assigned to the rule's VLAN, but not learned.
5. DHCP Generic	DHCP frame.	Frame source is assigned to the rule's VLAN, but not learned.
6. MAC Address	Frames contain a matching source MAC address.	Frame source is assigned to the rule's VLAN.
7. MAC Range	Frame contains a source MAC address that falls within a specified range of MAC addresses.	Frame source is assigned to the rule's VLAN.
8. Network Address	Frame contains a matching IP sub-net address, or	Frame source is assigned to the rule's VLAN.
9. Protocol	Frame contains a matching protocol type.	Frame source is assigned to the rule's VLAN.
10. Default	Frame does not match any rules.	Frame source is assigned to mobile port's default VLAN.

Configuring VLAN Rule Definitions

Note the following when configuring rules for a VLAN:

- The VLAN must already exist. Use the **vlan** command to create a new VLAN or the **show vlan** command to verify a VLAN is already configured. Refer to [Chapter 4, “Configuring VLANs,”](#) for more information.
- Which type of rule is needed; DHCP, MAC address, protocol, network address, or port. Refer to [“VLAN Rule Types” on page 8-4](#) for a summary of rule type definitions.
- IP network address rules are applied to traffic received on both mobile *and* fixed ports. If traffic contains a source IP address that is included in the subnet specified by the rule, the traffic is dropped. This does not occur, however, if the IP network address rule is configured on the default VLAN for the fixed port.
- If mobile port traffic matches rules defined for more than one VLAN, the mobile port is dynamically assigned to the VLAN with the higher precedence rule. Refer to [“Understanding VLAN Rule Precedence” on page 8-6](#) for more information.
- It is possible to define multiple rules for the same VLAN, as long as each rule is different. If mobile port traffic matches only one of the rules, the port and traffic are dynamically assigned to that VLAN.
- There is no limit to the number of rules defined for a single VLAN and up to 8129 rules are allowed per switch.
- It is possible to create a protocol rule based on Ether type, SNAP type, or DSAP/SSAP values. However, using predefined rules (such as MAC address, network address, and generic protocol rules) is recommended to ensure accurate results when capturing mobile port traffic.
- When an active device is disconnected from a mobile port and connected to a fixed port, the source MAC address of that device is not learned on the fixed port until the MAC address has aged out and no longer appears on the mobile port.
- When a VLAN is administratively disabled, static port and dynamic mobile port assignments are retained but traffic on these ports is not forwarded. However, VLAN rules remain active and continue to classify mobile port traffic for VLAN membership.
- When a VLAN is deleted from the switch configuration, all rules defined for that VLAN are automatically removed and any static or dynamic port assignments are dropped.

Refer to the following sections (listed in the order of rule precedence) for instructions on how to define each type of VLAN rule:

Rule	See
DHCP MAC Address	“Defining DHCP MAC Address Rules” on page 8-9
DHCP MAC Range	“Defining DHCP MAC Range Rules” on page 8-9
DHCP Port	“Defining DHCP Port Rules” on page 8-10
DHCP Generic	“Defining DHCP Generic Rules” on page 8-10
MAC Address	“Defining MAC Address Rules” on page 8-10
MAC Address Range	“Defining MAC Range Rules” on page 8-11
Network Address	“Defining IP Network Address Rules” on page 8-11 and “Defining Protocol Rules” on page 8-12

Rule	See
Protocol	“Defining Protocol Rules” on page 8-12
Port	“Defining Port Rules” on page 8-13

To display a list of VLAN rules already configured on the switch, use the **show vlan rules** command. For more information about this command, refer to the *OmniSwitch 6250 CLI Reference Guide*.

Defining DHCP MAC Address Rules

DHCP MAC address rules capture DHCP frames that contain a source MAC address that matches the MAC address specified in the rule. See [“Application Example: DHCP Rules” on page 8-14](#) for an example of how DHCP port rules are used in a typical network configuration.

To define a DHCP MAC address rule, enter **vlan** followed by an existing VLAN ID then **dhcp mac** followed by a valid MAC address. For example, the following command defines a DHCP MAC address rule for VLAN 255:

```
-> vlan 255 dhcp mac 00:00:da:59:0c:11
```

Only one MAC address is specified when using the **vlan dhcp mac** command to create a DHCP MAC rule. Therefore, to specify multiple MAC addresses for the same VLAN, create a DHCP MAC rule for each address. If dealing with a large number of MAC addresses in sequential order, consider using a DHCP MAC range rule described in the next section.

Use the **no** form of the **vlan dhcp mac** command to remove a DHCP MAC address rule.

```
-> vlan 255 no dhcp mac 00:00:da:59:0c:11
```

Defining DHCP MAC Range Rules

A DHCP MAC range rule is similar to a DHCP MAC address rule, but allows the user to specify a range of MAC addresses. This is useful when it is necessary to define rules for a large number of sequential MAC addresses. One DHCP MAC range rule could serve the same purpose as 10 or 20 DHCP MAC address rules, requiring less work to configure.

DHCP frames that contain a source MAC address that matches the low or high end MAC or that falls within the range specified by the low and high end MAC trigger dynamic port assignment to the rule’s VLAN. To define a DHCP MAC range rule, enter **vlan** followed by an existing VLAN ID then **dhcp mac range** followed by valid low and high end MAC addresses. For example, the following command creates a DHCP MAC range rule for VLAN 1100:

```
-> vlan 1100 dhcp mac range 00:00:da:00:00:01 00:00:da:00:00:09
```

Only valid source MAC addresses are allowed for the low and high end boundary MACs. For example, multicast addresses (e.g., 01:00:00:c5:09:1a) are ignored even if they fall within a specified MAC range and are not allowed as the low or high end boundary MAC. If an attempt is made to use a multicast address for one of the boundary MACs, an error message is displayed and the rule is not created.

Use the **no** form of the **vlan dhcp mac range** command to remove a DHCP MAC range rule. Note that it is only necessary to enter the low end MAC address to identify which rule to remove.

```
-> vlan 1000 no dhcp mac range 00:00:da:00:00:01
```

Defining DHCP Port Rules

DHCP port rules capture DHCP frames that are received on a mobile port that matches the port specified in the rule. See [“Application Example: DHCP Rules” on page 8-14](#) for an example of how DHCP port rules are used in a typical network configuration.

To define a DHCP port rule, enter **vlan** followed by an existing VLAN ID then **dhcp port** followed by a slot/port designation. For example, the following command defines a DHCP port rule for VLAN 255:

```
-> vlan 255 dhcp port 2/3
```

To specify multiple ports and/or slots, use a hyphen to specify a range of ports and a space to specify multiple slots. For example,

```
-> vlan 255 dhcp port 4/1-5 5/12-20 6/10-15
```

Use the **no** form of the **vlan dhcp port** command to remove a DHCP port rule.

```
-> vlan 255 no dhcp port 2/10-12 3/1-5 6/1-9
```

Defining DHCP Generic Rules

DHCP generic rules capture all DHCP traffic that does not match an existing DHCP MAC or DHCP port rule. If none of these other rules exist, then all DHCP frames are captured regardless of the port they came in on or the frame's source MAC address. Only one rule of this type is allowed per switch.

To define a DHCP generic rule, enter **vlan** followed by an existing VLAN ID then **dhcp generic**. For example,

```
-> vlan 255 dhcp generic
```

Use the **no** form of the **vlan dhcp generic** command to remove a DHCP generic rule.

```
-> vlan 255 no dhcp generic
```

Defining MAC Address Rules

MAC address rules capture frames that contain a source MAC address that matches the MAC address specified in the rule. The mobile port that receives the matching traffic is dynamically assigned to the rule's VLAN. Using MAC address rules, however, limits dynamic port assignment to a single VLAN. A mobile port can only belong to one MAC address rule VLAN, even if it sends traffic that matches rules defined for other VLANs.

For example, if VLAN 10 has a MAC address rule defined for 00:00:2a:59:0c:f1 and VLAN 20 has an IP protocol rule defined, mobile port 4/2 sending IP traffic with a source MAC address of 00:00:2a:59:0c:f1 is only assigned to VLAN 10. All mobile port 4/2 traffic is forwarded on VLAN 10, even though its traffic also matches the VLAN 20 IP protocol rule.

To define a MAC address rule, enter **vlan** followed by an existing VLAN ID then **mac** followed by a valid MAC address. For example, the following command defines a MAC address rule for VLAN 255:

```
-> vlan 255 mac 00:00:da:59:0c:11
```

Only one MAC address is specified when using the **vlan mac** command to create a MAC address rule. Therefore, to specify multiple MAC addresses for the same VLAN, create a separate rule for each address. If dealing with a large number of MAC addresses, consider using MAC address range rules described in the next section.

Use the **no** form of the **vlan mac** command to remove a MAC address rule.

```
-> vlan 255 no mac 00:00:da:59:0c:11
```

Defining MAC Range Rules

A MAC range rule is similar to a MAC address rule, but allows the user to specify a range of MAC addresses. This is useful when it is necessary to define rules for a large number of sequential MAC addresses. One MAC range rule could serve the same purpose as 10 or 20 MAC address rules, requiring less work to configure.

Frames that contain a source MAC address that matches the low or high end MAC or that falls within the range specified by the low and high end MAC trigger dynamic port assignment to the rule's VLAN. As is the case with MAC address rules, dynamic port assignment is limited to a single VLAN. A mobile port can only belong to one MAC range rule VLAN, even if it sends traffic that matches rules defined for other VLANs.

To define a MAC range rule, enter **vlan** followed by an existing VLAN ID then **mac range** followed by valid low and high end MAC addresses. For example, the following command creates a MAC range rule for VLAN 1000:

```
-> vlan 1000 mac range 00:00:da:00:00:01 00:00:da:00:00:09
```

Only valid source MAC addresses are allowed for the low and high end boundary MACs. For example, multicast addresses (e.g., 01:00:00:c5:09:1a) are ignored even if they fall within a specified MAC range and are not allowed as the low or high end boundary MAC. If an attempt is made to use a multicast address for one of the boundary MACs, an error message is displayed and the rule is not created.

Use the **no** form of the **vlan mac range** command to remove a MAC range rule. Note that it is only necessary to enter the low end MAC address to identify which rule to remove.

```
-> vlan 1000 no mac range 00:00:da:00:00:01
```

Defining IP Network Address Rules

IP network address rules capture frames that contain a source IP subnet address that matches the IP subnet address specified in the rule. If DHCP is used to provide client workstations with an IP address, consider using one of the DHCP rules in combination with an IP network address rule. See [“Application Example: DHCP Rules” on page 8-14](#) for an example of how IP network address and DHCP rules are used in a typical network configuration.

Note. IP network address rules are applied to traffic received on both mobile *and* fixed (non-mobile) ports. As a result, fixed port traffic that contains an IP address that is included in the IP subnet specified by the rule is dropped. However, if the IP network address rule VLAN is also the default VLAN for the fixed port, then the fixed port traffic is forwarded and not dropped.

To define an IP network address rule, enter **vlan** followed by an existing VLAN ID then **ip** followed by a valid IP network address and an optional subnet mask. For example, the following command creates an IP network address rule for VLAN 1200:

```
-> vlan 1200 ip 31.0.0.0 255.0.0.0
```

In this example, frames received on any mobile port must contain a network 31.0.0.0 source IP address (e.g., 31.0.0.10, 31.0.0.4) to qualify for dynamic assignment to VLAN 1200.

If a subnet mask is not specified, the default class for the IP address is used (Class A, B, or C). For example, either one of the following commands will create an IP network address rule for network 134.10.0.0:

```
-> vlan 1200 ip 134.10.0.0 255.255.0.0
-> vlan 1200 ip 134.10.0.0
```

The pool of available internet IP addresses is divided up into three classes, as shown in the following table. Each class includes a range of IP addresses. The range an IP network address belongs to determines the default class for the IP network when a subnet mask is not specified.

Network Range	Class
1.0.0.0 - 126.0.0.0	A
128.1.0.0 - 191.254.0.0	B
192.0.1.0 - 223.255.254.0	C

Use the **no** form of the **vlan ip** command to remove an IP network address rule.

```
-> vlan 1200 no ip 134.10.0.0
```

Defining Protocol Rules

Protocol rules capture frames that contain a protocol type that matches the protocol value specified in the rule. There are several generic protocol parameter values to select from; IP Ethernet-II, IP SNAP, Ethernet II, DECNet, and AppleTalk. If none of these are sufficient to capture the desired type of traffic, use the Ethertype, DSAP/SSAP, or SNAP parameters to define a more specific protocol type value.

To define a protocol rule, enter **vlan** followed by an existing VLAN ID then **protocol** followed by a valid protocol parameter value. For example, the following commands define a protocol rule for VLAN 1503 and VLAN 1504:

```
-> vlan 1503 protocol ip-snap
-> vlan 1504 protocol dsapssap f0/f0
```

The first example command specifies that frames received on any mobile port must contain an IP SNAP protocol type to qualify for dynamic assignment to VLAN 1503. The second command specifies that frames received on any mobile port must contain a DSAP/SSAP protocol value of f0/f0 to qualify for dynamic assignment to VLAN 1504.

If an attempt is made to define an ethertype rule with a protocol type value that is equal to the value already captured by one of the generic IP protocol rule, a message displays recommending the use of the IP generic rule. The following example shows what happens when an attempt is made to create a protocol rule with an ethertype value of 0800 (IP Ethertype):

```
-> vlan 200 protocol ethertype 0800
ERROR: Part of ip ethernet protocol class - use <vlan # protocol ip-e2> instead
```

The following table lists keywords for specifying a protocol type:

protocol type keywords	
ip-e2	ethertype
ip-snap	dsapssap
decnet	snap
appletalk	

Note that specifying a SNAP protocol type restricts classification of mobile port traffic to the ethertype value found in the IEEE 802.2 SNAP LLC frame header.

Use the **no** form of the **vlan protocol** command to remove a protocol rule.

```
-> vlan 1504 no protocol dsapssap f0/f0
```

Defining Port Rules

Port rules do not require mobile port traffic to trigger dynamic assignment. When this type of rule is defined, the specified mobile port is immediately assigned to the specified VLAN. As a result, port rules are often used for silent network devices, which do not trigger dynamic assignment because they do not send traffic.

Port rules only apply to outgoing mobile port broadcast types of traffic and do not classify incoming traffic. In addition, multiple VLANs can have the same port rule defined. The advantage to this is that broadcast traffic from multiple VLANs is forwarded out one physical mobile port. When a mobile port is specified in a port rule, however, its incoming traffic is still classified for VLAN assignment in the same manner as all other mobile port traffic.

To define a port rule, enter **vlan** followed by an existing VLAN ID then **port** followed by a mobile **slot/port** designation. For example, the following command creates a port rule for VLAN 755:

```
-> vlan 755 port 2/3
```

In this example, all traffic on VLAN 755 is flooded out mobile port 2 on slot 3.

Note that it is possible to define a port rule for a non-mobile (fixed, untagged) port, however, the rule is not active until mobility is enabled on the port.

Use the **no** form of the **vlan port** command to remove a port rule.

```
-> vlan 755 no port 2/3
```

Application Example: DHCP Rules

This application example shows how Dynamic Host Configuration Protocol (DHCP) port and MAC address rules are used in a DHCP-based network. DHCP is built on a client-server model in which a designated DHCP server allocates network addresses and delivers configuration parameters to dynamically configured clients.

Since DHCP clients initially have no IP address, assignment of these clients to a VLAN presents a problem. The switch determines VLAN membership by looking at traffic from source devices. Since the first traffic transmitted from a source DHCP client does not contain the actual address for the client (because the server has not allocated the address yet), the client may not have the same VLAN assignment as its server.

Before the introduction of DHCP port and MAC address rules, various strategies were deployed to use DHCP with VLANs. Typically these strategies involved IP protocol and network address rules along with DHCP Relay functionality. These solutions required the grouping of all DHCP clients in a particular VLAN through a common IP policy.

DHCP port and MAC address rules simplify the configuration of DHCP networks. Instead of relying on IP-based rules to group all DHCP clients in the same network as a DHCP server, you can manually place each individual DHCP client in the VLAN or mobile group of your choice.

The VLANs

This application example contains three (3) VLANs. These VLANs are called Test, Production, and Branch. The Test VLAN connects to the main network, the Production VLAN, through an external router. The configuration of this VLAN is self-contained, making it easy to duplicate for testing purposes. The Test VLAN contains its own DHCP server and DHCP clients. The clients gain membership to the VLAN through DHCP port rules.

The Production VLAN carries most of the traffic in this network. It does not contain a DHCP server, but does contain DHCP clients that gain membership through DHCP port rules. Two external routers connect this VLAN to the Test VLAN and a Branch VLAN. One of the external routers—the one connected to the Branch VLAN—has DHCP Relay functionality enabled. It is through this router that the DHCP clients in the Production VLAN access the DHCP server in the Branch VLAN.

The Branch VLAN contains a number of DHCP client stations and its own DHCP server. The DHCP clients gain membership to the VLAN through both DHCP port and MAC address rules. The DHCP server allocates IP addresses to all Branch and Production VLAN clients.

DHCP Servers and Clients

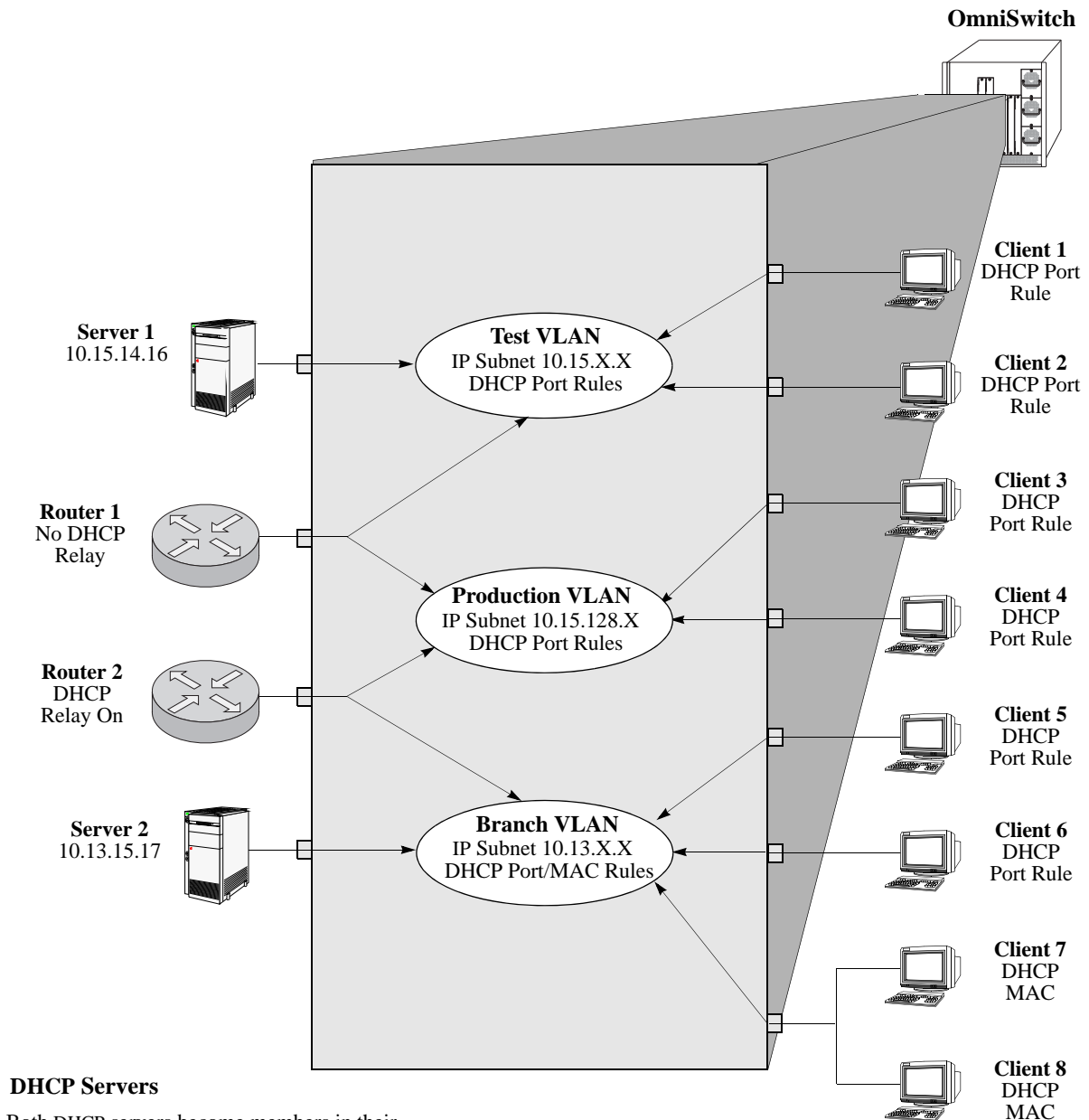
DHCP clients must communicate with a DHCP server at initialization. The most reliable way to ensure this communication is for the server and its associated clients to share the same VLAN. However, if the network configuration does not lend itself to this solution (as the Production VLAN does not in this application example), then the server and clients can communicate through a router with DHCP Relay enabled.

The DHCP servers and clients in this example are either in the same VLAN or are connected through a router with DHCP Relay. All clients in the Test VLAN receive IP addresses from the server in their VLAN (Server 1). Likewise, all clients in the Branch VLAN receive IP addresses from their local server (Server 2). The DHCP clients in the Production VLAN do not have a local DHCP server, so they must rely on the DHCP Relay functionality in external Router 2 to obtain their IP addresses from the DHCP server in the Branch VLAN.

Both DHCP servers are assigned to their VLANs through IP network address rules.

The following table summarizes the VLAN architecture and rules for all devices in this network configuration. The diagram on the following page illustrates this network configuration.

Device	VLAN Membership	Rule Used/Router Role
DHCP Server 1	Test VLAN	IP network address rule=10.15.0.0
DHCP Server 2	Branch VLAN	IP network address rule=10.13.0.0
External Router 1	Test VLAN Production VLAN	Connects Test VLAN to Production VLAN
External Router 2	Production VLAN Branch VLAN	DHCP Relay provides access to DHCP server in Branch VLAN for clients in Production VLAN.
DHCP Client 1	Test VLAN	DHCP Port Rule
DHCP Client 2	Test VLAN	DHCP Port Rule
DHCP Client 3	Production VLAN	DHCP Port Rule
DHCP Client 4	Production VLAN	DHCP Port Rule
DHCP Client 5	Branch VLAN	DHCP Port Rule
DHCP Client 6	Branch VLAN	DHCP Port Rule
DHCP Client 7	Branch VLAN	DHCP MAC Address Rule
DHCP Client 8	Branch VLAN	DHCP MAC Address Rule



DHCP Servers

Both DHCP servers become members in their respective VLANs via IP subnet rules.

Routers

Router 1 provides connectivity between the Test VLAN and the Production VLAN. It does not have Bootup functionality enabled so it cannot connect DHCP servers and clients from different VLANs.

Router 2 connects the Production VLAN and the Branch VLAN. With DHCP Relay enabled, this router can provide connectivity between the DHCP server in the Branch VLAN and the DHCP clients in the Production VLAN.

DHCP Clients

Clients 1 to 6 are assigned to their respective VLANs through DHCP port rules. Clients 3 and 4 are not in a VLAN with a DHCP server so they must rely on the server in the Branch VLAN for initial addressing information. Clients 7 and 8 share a port with other devices, so they are assigned to the Branch VLAN via DHCP MAC address rules.

DHCP Port and MAC Rule Application Example

Verifying VLAN Rule Configuration

To display information about VLAN rules configured on the switch, use the following **show** command;

show vlan rules Displays a list of rules for one or all VLANs configured on the switch.

For more information about the resulting display from this command, see the *OmniSwitch 6250 CLI Reference Guide*. An example of the output for the **show vlan rules** command is also given in “[Sample VLAN Rule Configuration](#)” on page 8-3.

9 Configuring VLAN Stacking

VLAN Stacking provides a mechanism to tunnel multiple customer VLANs (CVLAN) through a service provider network using one or more service provider VLANs (SVLAN) by way of 802.1Q double-tagging or VLAN Translation. This feature enables service providers to offer their customers Transparent LAN Services (TLS). This service is multipoint in nature so as to support multiple customer sites or networks distributed over the edges of a service provider network.

This implementation of VLAN Stacking offers the following functionality:

- An Ethernet service-based approach that is similar to configuring a virtual private LAN service (VPLS).
- Ingress bandwidth sharing across User Network Interface (UNI) ports.
- Ingress bandwidth rate limiting on a per UNI port, per CVLAN, or CVLAN per UNI port basis.
- CVLAN (inner) tag 802.1p-bit mapping to SVLAN (outer) tag 802.1p bit.
- CVLAN (inner) tag DSCP mapping to SVLAN (outer) tag 802.1p bit.
- Profiles for saving and applying traffic engineering parameter values.

In This Chapter

This chapter describes the basic components of VLAN Stacking and how to define a service-based or port-based configuration through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

This chapter provides an overview of VLAN Stacking and includes the following topics:

- [“VLAN Stacking Specifications” on page 9-2.](#)
- [“VLAN Stacking Defaults” on page 9-2.](#)
- [“VLAN Stacking Overview” on page 9-3.](#)
- [“Interaction With Other Features” on page 9-7.](#)
- [“Configuring VLAN Stacking Services” on page 9-11](#)
- [“VLAN Stacking Application Examples” on page 9-22.](#)
- [“Wire-Speed Ethernet Loopback Test” on page 9-25.](#)

VLAN Stacking Specifications

IEEE Standards Supported	IEEE 802.1Q, 2003 Edition, IEEE Standards for Local and metropolitan area networks—Virtual Bridged Local Area Networks <i>P802.1ad/D6.0 (C/LM) Standard for Local and Metropolitan Area Networks - Virtual Bridged Local Area Networks - Amendment 4: Provider Bridges</i>
Platforms Supported	OmniSwitch 6250-Metro Models
Maximum number of SVLANs	4093 (VLAN 2 through 4094)
Maximum number of UNI port associations with CVLANs.	256
Features <i>not</i> supported on VLAN Stacking ports	Group Mobility, Authentication, and L3 Routing

VLAN Stacking Defaults

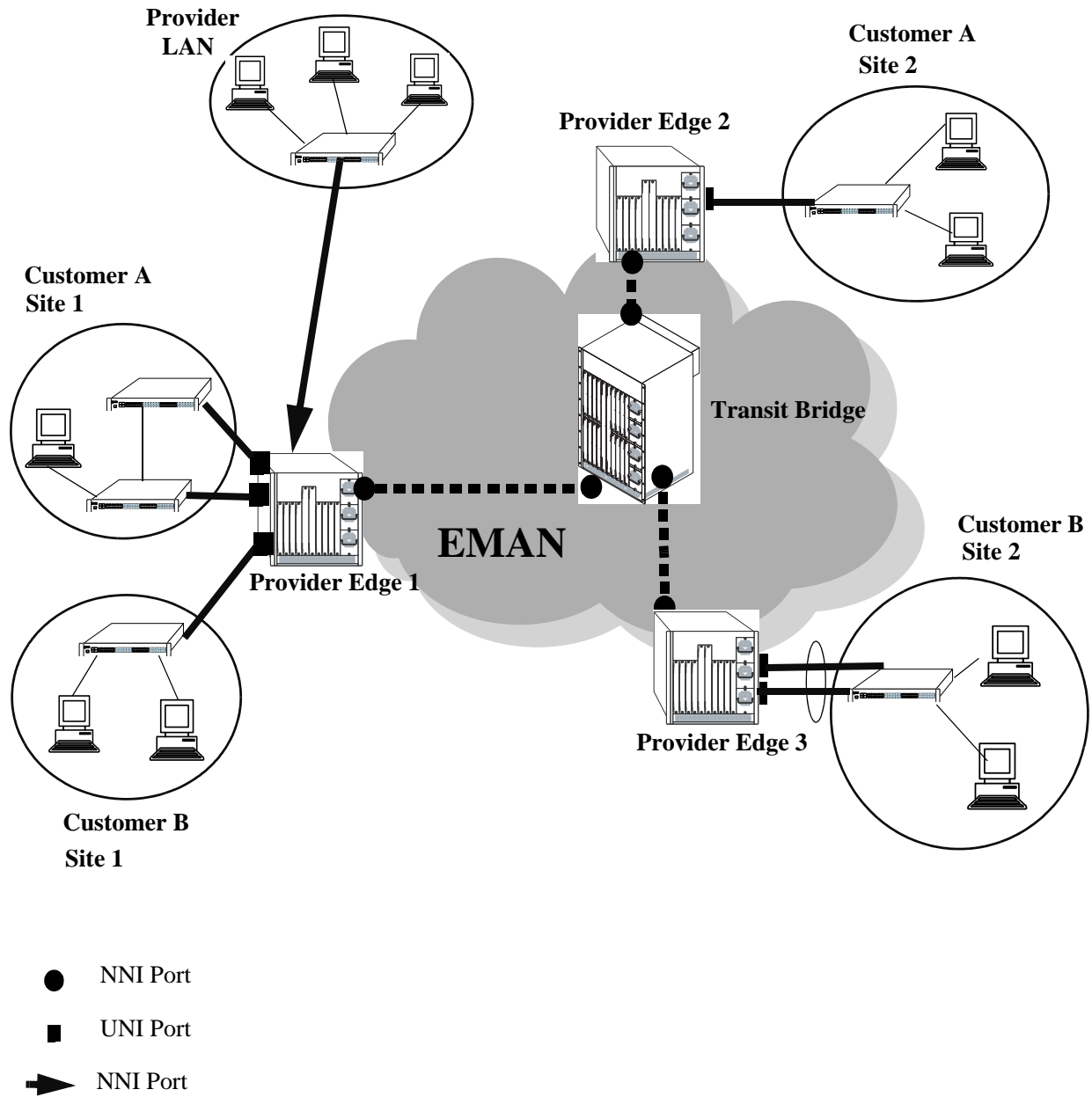
Parameter Description	Command	Default Value/Comments
SVLAN administrative and Spanning Tree status.	ethernet-service svlan	Enabled
IPMVLAN administrative and Spanning Tree status.	ethernet-service ipmvlan	Enabled
Vendor TPID and legacy BPDU support for STP or GVRP on a VLAN Stacking network port.	ethernet-service nni	TPID = 0x8100 legacy STP BPDU = dropped. legacy GVRP BPDU = dropped.
Acceptable frame types on a VLAN Stacking user port.	ethernet-service sap cvlan	None.
Traffic engineering profile attributes for a VLAN Stacking Service Access Point (SAP).	ethernet-service sap-profile	ingress bandwidth = shared ingress bandwidth mbps = 0 CVLAN tag is preserved. SVLAN priority mapping = 0
Treatment of customer protocol control frames ingressing on a VLAN Stacking user port.	ethernet-service uni-profile	Processed Frames: 802.3ad, UDLD, OAM, LACP-Marker Tunneled Frames: STP, GVRP Discarded Frames: 802.1x, 802.1ab, AMAP, VTP VLAN, Uplink Fast, PVST, PAGP, DTP, CDP

VLAN Stacking Overview

VLAN Stacking provides a mechanism for defining a transparent bridging configuration through a service provider network. The major components of VLAN Stacking that provide this type of functionality are described as follows:

- **Provider Edge (PE) Bridge**—An ethernet switch that resides on the edge of the service provider network. The PE Bridge interconnects customer network space with service provider network space. A switch is considered a PE bridge if it transports packets between a customer-facing port and a network port or between two customer-facing ports.
- **Transit Bridge**—An ethernet switch that resides inside the service provider network and provides a connection between multiple provider networks. It employs the same SVLAN on two or more network ports. This SVLAN does not terminate on the switch itself; traffic ingressing on a network port is switched to other network ports. It is also possible for the same switch to function as a both a PE Bridge and a Transit Bridge.
- **Tunnel (SVLAN)**—A tunnel, also referred to as an SVLAN, is a logical entity that connects customer networks by transparently bridging customer traffic through a service provider network. The tunnel is defined by an SVLAN tag that is appended to all customer traffic. This implementation provides the following three types of SVLANs, which are both defined by the type of traffic that they carry:
 - an SVLAN that *carries customer traffic*
 - an SVLAN that *carries provider management traffic*
 - an IP Multicast VLAN (IPMVLAN) that *distributes multicast traffic*
- **Network Network Interface (NNI)**—An NNI is a port that resides on either a PE Bridge or a Transit Bridge and connects to a service provider network. Traffic ingressing on a network port is considered SVLAN traffic and is switched to a customer-facing port or to another network port.
- **User Network Interface (UNI)**—A UNI is a port that resides on a PE bridge that connects to a customer network and carries customer traffic. The UNI may consist of a single port or an aggregate of ports and can accept tagged or untagged traffic.

The following illustration shows how VLAN Stacking uses the above components to tunnel customer traffic through a service provider network:



VLAN Stacking Elements

How VLAN Stacking Works

On the Provider Edge bridge (PE), a unique tunnel (SVLAN) ID is assigned to each customer. The tunnel ID corresponds to a VLAN ID, which is created on the switch when the tunnel is configured. For example, when tunnel 100 is created, VLAN Stacking software interacts with VLAN Manager software to configure a VLAN 100 on the switch. VLAN 100 is the provider bridge VLAN that will tunnel customer VLAN traffic associated with tunnel 100. So, there is a one to one correspondence between a tunnel and its provider bridge VLAN ID. In fact, tunnel and VLAN are interchangeable terms when referring to the provider bridge configuration.

VLAN Stacking refers to the tunnel encapsulation process of appending to customer packets an 802.1Q tag that contains the tunnel ID associated to that customer's provider bridge port and/or VLANs. The encapsulated traffic is then transmitted through the Ethernet metro area network (EMAN) cloud and received on another PE bridge that contains the same tunnel ID, where the packet is then stripped of the tunnel tag and forwarded to the traffic destination.

The following provides an example of how a packet ingressing on a VLAN Stacking UNI port that is tagged with the customer VLAN (CVLAN) ID transitions through the VLAN Stacking encapsulation process:

- 1 Packet with CVLAN tag ingressing on a user port.

MAC DA (6)	MAC SA (6)	CVLAN Tag (4)	ETYPE 0x0800	Payload
---------------	---------------	------------------	-----------------	---------

- 2 **Double Tagging** inserts the SVLAN tag in the packet. The packet is sent out the network port with double tags (SVLAN+CVLAN).

MAC DA (6)	MAC SA (6)	SVLAN Tag (4)	CVLAN Tag (4)	ETYPE 0x0800	Payload
---------------	---------------	------------------	------------------	-----------------	---------

- 3 **VLAN Translation** replaces the CVLAN Tag with SVLAN Tag. The packet is sent out the network port with a single tag (SVLAN).

MAC DA (6)	MAC SA (6)	SVLAN Tag (4)	ETYPE 0x0800	Payload
---------------	---------------	------------------	-----------------	---------

VLAN Stacking Services

The VLAN Stacking application uses an Ethernet service based approach for tunneling customer traffic through a provider network. This approach requires the configuration of the following components to define a tunneling service:

- **VLAN Stacking Service**—A service name that is associated with an SVLAN, NNI ports, and one or more VLAN Stacking service access points. The service identifies the customer traffic that the SVLAN will carry through the provider traffic.
- **Service Access Point (SAP)**—A SAP is associated with a VLAN Stacking service name and a SAP profile. The SAP binds UNI ports and customer traffic received on those ports to the service. The profile specifies traffic engineering attribute values that are applied to the customer traffic received on the SAP UNI ports.
- **Service Access Point (SAP) Profile**—A SAP profile is associated with a SAP ID. Profile attributes define values for ingress bandwidth sharing, rate limiting, CVLAN tag processing (translate or preserve), and priority mapping (inner to outer tag or fixed value).
- **UNI Port Profile**—This type of profile is associated with each UNI port and configures how Spanning Tree and GVRP control packets are processed on the UNI port.

See the [“Configuring VLAN Stacking Services” on page 9-11](#) for more information.

Interaction With Other Features

This section contains important information about VLAN Stacking interaction with other OmniSwitch features. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

GARP VLAN Registration Protocol (GVRP)

- GVRP control frames are tunneled by default; processing of GVRP frames similar to processing of Spanning Tree frames (see below).
- The VLAN Stacking provider edge (PE) switch will not tunnel GVRP frames unless the GVRP feature and/or GVRP transparent switching functionality is enabled on the PE switch. This is true even if GVRP processing is enabled for the VLAN Stacking port.

IP Multicast VLANs

The IP Multicast VLANs (IPMV) application has the following interactions with VLAN Stacking functionality and commands:

- IPMV operates in one of two modes: enterprise or VLAN Stacking. When the enterprise mode is active, IPMV uses sender and receiver ports for IP multicast traffic. When the IPMV VLAN Stacking mode is active, IPMV maps sender and receiver ports to VLAN Stacking NNI and UNI ports.
- If IPMV is operating in the enterprise mode, there are no CLI usage changes.
- If IPMV is operating in the VLAN Stacking mode, the following VLAN Stacking CLI commands are used to configure interoperability with IPMV:

VLAN Stacking Commands

[ethernet-service ipmvlan](#)

[ethernet-service svlan nni](#)

[ethernet-service sap](#)

[ethernet-service sap uni](#)

[ethernet-service sap cvlan](#)

[vlan ipmvlan ctag](#)

[vlan ipmvlan address](#)

[vlan ipmvlan sender-port](#)

[vlan ipmvlan receiver-port](#)

[ethernet-service sap-profile](#)

[ethernet-service sap sap-profile](#)

See the *OmniSwitch 6250 CLI Reference Guide* for more information about these commands.

Link Aggregation

- Both static and dynamic link aggregation are supported with VLAN Stacking.
- Note that a link aggregate must consist of all UNI or all NNI ports. VLAN Stacking functionality is not supported on link aggregates that consist of a mixture of VLAN Stacking ports and conventional switch ports.

Quality of Service (QoS)

The QoS application has the following interactions with VLAN Stacking:

- QoS policy rules take precedence over the VLAN Stacking SAP profile configuration. As a result, it is possible to configure QoS policy rules to override VLAN Stacking SAP profile settings, such as bandwidth and priority.
- VLAN Stacking ports are trusted and use 802.1p classification by default.
- QoS applies the **source vlan** and **802.1p** policy conditions to the SVLAN (outer) tag of VLAN Stacking packets.

Ring Rapid Spanning Tree Protocol (RRSTP)

- RRSTP is only supported on VLAN Stacking NNI ports; UNI ports are not supported.
- An RRSTP ring must consist of either all VLAN Stacking NNI ports or all standard switch ports; a mixture of the two port types in the same ring is not supported.
- If an RRSTP ring contains NNI ports, the VLAN tag configured for the ring must match the SVLAN tag that VLAN Stacking appends to packets before they are received or forwarded on NNI ports.

Spanning Tree

- Spanning Tree is enabled by default for VLAN Stacking SVLANs. The Spanning Tree status for an SVLAN is configurable through VLAN Stacking commands. Note that the SVLAN Spanning Tree status applies only to the service provider network topology.
- BPDU frames are tunneled by default. See [“Configuring a UNI Profile” on page 9-20](#) for information about configuring VLAN Stacking to tunnel or discard Spanning Tree BPDU.
- See [“Configuring VLAN Stacking Network Ports” on page 9-14](#) for information about configuring VLAN Stacking interoperability with *legacy* Spanning Tree BPDU systems.
- A back door link configuration is not supported. This occurs when there is a link between two customer sites that are both connected to a VLAN Stacking provider edge switch.
- A dual home configuration is not supported. This type of configuration consists of a single customer site connected to two different VLAN Stacking switches or two switches at a customer site connect to two different VLAN Stacking switches.

Quick Steps for Configuring VLAN Stacking

The following steps provide a quick tutorial for configuring a VLAN Stacking service:

- 1 Create a VLAN Stacking VLAN (SVLAN) 1001 using the **ethernet-service** command.

```
-> ethernet-service svlan 1001
```

- 2 Create a VLAN Stacking service and associate the service with SVLAN 1001 using the **ethernet-service service-name** command.

```
-> ethernet-service service-name CustomerA svlan 1001
```

- 3 Configure port 3/1 as a VLAN Stacking Network Network Interface (NNI) port and associate the port with SVLAN 1001 using the **ethernet-service svlan nni** command.

```
-> ethernet-service svlan 1001 nni 3/1
```

- 4 Create a VLAN Stacking Service Access Point (SAP) and associate it to the “CustomerA” service using the **ethernet-service sap** command.

```
-> ethernet-service sap 10 service-name CustomerA
```

- 5 Configure port 1/49 as a VLAN Stacking User Network Interface (UNI) port and associate the port with SAP ID 10 using the **ethernet-service sap uni** command.

```
-> ethernet-service sap 10 uni 1/49
```

- 6 Associate traffic from customer VLANs (CVLAN) 10 and 20 with SAP 10 using the **ethernet-service sap cvlan** command.

```
-> ethernet-service sap 10 cvlan 10  
-> ethernet-service sap 10 cvlan 20
```

- 7 (Optional) Create a SAP profile that applies an ingress bandwidth of 10, translates the CVLAN tag, and maps the CVLAN priority to the SVLAN priority using the **ethernet-service sap-profile** command.

```
-> ethernet-service sap-profile sap-video1 ingress-bandwidth 10 cvlan translate  
priority map-inner-to-outer-p
```

- 8 (Optional) Associate the “sap-video1” profile with SAP 10 using the **ethernet-service sap sap-profile** command.

```
-> ethernet-service sap 10 sap-profile sap-video1
```

- 9 (Optional) Create a UNI port profile to block GVRP and STP control frames received on UNI ports using the **ethernet-service uni-profile** command.

```
-> ethernet-service uni-profile uni_1 l2-protocol stp gvrp discard
```

- 10 (Optional) Associate the “uni_1” profile with port 1/49 using the **ethernet-service uni uni-profile** command.

```
-> ethernet-service uni 1/49 uni-profile uni_1
```

Note. Verify the VLAN Stacking Ethernet service configuration using the [show ethernet-service](#) command:

```
-> show ethernet-service

Service Name : CustomerA
  SVLAN      : 1001
  NNI(s)     : 3/1
  SAP Id     : 10
    UNIs      : 1/49
    CVLAN(s)  : 10, 20
    sap-profile : sap-video1

Service Name : ipmv_service
  IPMVLAN    : 40
  NNI(s)     : No NNIs configured
  SAP Id     : 2
    UNIs      : 1/22
    CVLAN(s)  : 100
    sap-profile : translate_profile

Service Name : Video-Service
  SVLAN      : 300
  NNI(s)     : 2/1, 3/2
  SAP Id     : 20
    UNIs      : 1/1, 1/2
    CVLAN(s)  : 10, 20
    sap-profile : sap-video1
  SAP Id     : 30
    UNIs      : 1/3
    CVLAN(s)  : untagged, 40
    sap-profile : sap-video2
```

See the *OmniSwitch 6250 CLI Reference Guide* for information about the fields in this display.

Configuring VLAN Stacking Services

Configuring a VLAN Stacking Ethernet service requires several steps. These steps are outlined here and further described throughout this section. For a brief tutorial on configuring a VLAN Stacking service, see [“Quick Steps for Configuring VLAN Stacking” on page 9-9](#).

1 Create an SVLAN. An SVLAN is associated to a VLAN Stacking service to carry customer or provider traffic. In addition, an SVLAN may also distribute IP multicast traffic, if it is configured as an IP multicast VLAN (IPMVLAN). See [“Configuring SVLANs” on page 9-12](#).

2 Create a VLAN Stacking service. A service name is associated with an SVLAN to identify the customer traffic that the SVLAN will carry through the provider network. See [“Configuring a VLAN Stacking Service” on page 9-13](#).

3 Configure Network Network Interface (NNI) ports. An NNI port is associated with an SVLAN and carries the encapsulated SVLAN traffic through the provider network. See [“Configuring VLAN Stacking Network Ports” on page 9-14](#).

4 Configure a VLAN Stacking service access point (SAP). A SAP binds UNI ports, the type of customer traffic, and traffic engineering parameter attributes to the VLAN Stacking service. Each SAP is associated to one service name, but a single service can have multiple SAPs to which it is associated. See [“Configuring a VLAN Stacking Service Access Point” on page 9-15](#).

5 Configure User Network Interface (UNI) ports. One or more UNI ports are associated with a SAP to identify to the service which ports will receive customer traffic that the service will process for tunneling through the provider network. When a UNI port is associated with a SAP, the SAP parameter attributes are applied to traffic received on the UNI port. See [“Configuring VLAN Stacking User Ports” on page 9-16](#).

6 Associate CVLAN traffic with an SAP. This step specifies the type of traffic customer traffic that is allowed on UNI ports and then tunneled through the SVLAN. The type of customer traffic is associated with a SAP and applies to all UNI ports associated with the same SAP. See [“Configuring the Type of Customer Traffic to Tunnel” on page 9-17](#).

7 Define SAP profile attributes. A SAP profile contains traffic engineering attributes for specifying bandwidth sharing, rate limiting, CVLAN translation or double-tagging, and priority bit mapping. A default profile is automatically associated with a SAP at the time the SAP is created. As a result, it is only necessary to configure a SAP profile if the default attribute values are not sufficient. See [“Configuring a Service Access Point Profile” on page 9-18](#).

8 Define UNI profile attributes. A default UNI profile is automatically assigned to a UNI port at the time a port is configured as a VLAN Stacking UNI. This profile determines how control frames received on the port are processed. It is only necessary to configure a UNI profile if the default attribute values are not sufficient. See [“Configuring a UNI Profile” on page 9-20](#).

The following table provides a summary of commands used in these procedures:

Commands	Used for ...
ethernet-service	Creating SVLANs to tunnel customer or management traffic or an IP Multicast VLAN for distributing multicast traffic.
ethernet-service service-name	Creating a VLAN Stacking service and associating the service with an SVLAN or IP multicast VLAN.
ethernet-service svlan nni	Configuring a switch port as a VLAN Stacking NNI port and associating the NNI port with an SVLAN.
ethernet-service nni	Configuring a vendor TPID and legacy Spanning Tree or GVRP support for an NNI port.
ethernet-service sap	Creating a VLAN Stacking SAP and associates the SAP with a VLAN Stacking service name.
ethernet-service sap uni	Configuring a switch port as a VLAN Stacking UNI port and associating the UNI port with a VLAN Stacking SAP.
ethernet-service sap cvlan	Specifying the type of customer traffic that is accepted on SAP UNI ports.
ethernet-service sap-profile	Configures traffic engineering attributes for customer traffic that is accepted on SAP UNI ports.
ethernet-service sap sap-profile	Associates a VLAN Stacking SAP with a profile.
ethernet-service uni-profile	Configures how protocol control frames are processed on VLAN Stacking UNI ports.
ethernet-service uni uni-profile	Associates a VLAN Stacking UNI port with a profile.

Configuring SVLANs

There are three kinds of SVLANs: one that carries customer traffic, one that carries provider management traffic, and one that carries IP Multicast VLAN traffic (IPMVLAN). SVLANs are not configurable or modifiable using standard VLAN commands. The exception to this is that it is possible to configure an IP interface for a provider management SVLAN. However, traffic is not routed on this interface.

The **ethernet-service** command is used to create an SVLAN. This command provides parameters to specify the type of SVLAN: **svlan** (customer traffic), **management-vlan** (provider management traffic), or **ipmv** (IP Multicast traffic). For example, the following commands create a customer SVLAN, management SVLAN, and IP Multicast VLAN:

```
-> ethernet-service svlan 300
-> ethernet-service management-vlan 200
-> ethernet-service impv 500
```

Similar to standard VLANs, the administrative and Spanning Tree status for the SVLAN is enabled by default and the SVLAN ID is used as the default name. The **ethernet-service svlan** command also provides parameters for changing any of these status values and the name. These are the same parameters that are used to change these values for standard VLANs. For example, the following commands change the administrative and Spanning Tree status and name for SVLAN 300:

```
-> ethernet-service svlan 300 disable
-> ethernet-service svlan 300 stp disable
-> ethernet-service svlan 300 name "Customer A"
```

To delete an SVLAN from the switch configuration, use the **no** form of the **ethernet-service svlan** command. For example, to delete SVLAN 300 enter:

```
-> no ethernet-service svlan 300
```

Note that when an SVLAN is deleted, all port associations with the SVLAN are also removed.

Use the **show ethernet-service vlan** command to display a list of VLAN Stacking VLANs configured for the switch.

Configuring a VLAN Stacking Service

A VLAN Stacking service is identified by a name. The **ethernet-service service-name** command is used to create a service and assign the service to an SVLAN or IMPVLAN ID, depending on the type of traffic the service will process. The ID specified with this command identifies the SVLAN that will carry traffic for the service. Each service is associated with only one SVLAN, but an SVLAN may belong to multiple services.

To create a VLAN Stacking service, use the **ethernet-service service-name** command and specify a name and SVLAN or IMPVLAN ID. For example, the following command creates a service named “Video-Service” and associates the service with SVLAN 300:

```
-> ethernet-service service-name Video-Service svlan 300
```

The SVLAN or IMPVLAN ID specified with this command must already exist in the switch configuration; entering a standard VLAN ID is not allowed. See “[Configuring SVLANs](#)” on page 9-12 for more information.

Once the VLAN Stacking service is created, the name is used to configure and display all components associated with that service. The service name provides a single point of reference for a specific VLAN Stacking configuration. For example, the following **show ethernet-service** command display shows how the service name identifies a VLAN Stacking service and components related to that service:

```
->show ethernet-service
```

```
Service Name : Video-Service
  SVLAN      : 300
  NNI(s)    : 2/1, 3/2
  SAP Id    : 20
    UNIs     : 1/1, 1/2
    CVLAN(s) : 10, 20
    sap-profile : sap-video1
  SAP Id    : 30
    UNIs     : 1/3
    CVLAN(s) : untagged, 40
    sap-profile : sap-video2
Service Name : ipmv_service
  IMPVLAN   : 40
  NNI(s)    : No NNIs configured
  SAP Id    : 2
    UNIs     : 1/22
    CVLAN(s) : 100
    sap-profile : translate_profile
```

To delete a service from the switch configuration, use the **no** form of the **ethernet-service service-name** command. For example, the following command deletes the “Video-Service” service:

```
-> no ethernet-service servic-name Video-Service
```

Note that when a VLAN Stacking service is deleted, the SVLAN or IMPVLAN ID association with the service is automatically deleted. However, if one or more VLAN Stacking service access point (SAP) are associated with the service, remove the SAPs first before attempting to delete the service.

Configuring VLAN Stacking Network Ports

The **ethernet-service svlan nni** command is used to configure a switch port or link aggregate of ports as a VLAN Stacking Network Network Interface (NNI) and associate the NNI with an SVLAN. Note that NNI ports are not associated with IP Multicast VLANs. For example, the following command configures port 2/1 as an NNI port and associates 2/1 with SVLAN 300:

```
-> ethernet-service svlan 300 nni 2/1
```

When a port is associated with an SVLAN using this command, the port is automatically defined as an NNI to carry traffic for the specified SVLAN. In addition, the default VLAN for the port is changed to a VLAN that is reserved for the VLAN Stacking application. At this point, the port is no longer configurable using standard VLAN port commands.

To delete an NNI port association with an SVLAN, use the **no** form of the **ethernet-service svlan nni** command. For example, the following command deletes the association between NNI 2/1 and SVLAN 300:

```
-> no ethernet-service svlan 300 nni 2/1
```

Note that when the last SVLAN association for the port is deleted, the port automatically reverts back to a conventional switch port and is no longer VLAN Stacking capable.

Use the **show ethernet-service port** command to verify the NNI port configuration for the switch.

Configuring NNI Port Parameters

The **ethernet-service nni** command is used to configure the following parameters that apply to traffic processed by NNI ports:

- **tpid**—Configures the vendor TPID value for the SVLAN tag. This value is set to 0x8100 by default, and is applied to traffic egressing on the NNI port and is compared to the SVLAN tag of packets ingressing on the NNI port. If the configured NNI TPID value and the ingress packet value match, then the packet is considered an SVLAN tagged packet. If these values do not match, then the packet is classified as a non-SVLAN tagged packet.
- **gvrp legacy-bpdu**—Specifies whether or not legacy GVRP BPDU are tunneled on the NNI port. GVRP BPDU are dropped by default.
- **stp legacy-bpdu**—Specifies whether or not legacy Spanning Tree BPDU are tunneled on the NNI port. Spanning Tree BPDU are dropped by default.
- **transparent-bridging**—Configures the transparent bridging status for the NNI port. When transparent bridging is enabled, the NNI forwards SVLAN traffic without processing packet contents. As a result, the NNI port can also forward traffic for SVLANs that are not configured on the local switch, thus allowing for a greater number of NNI port associations with SVLANs. Enabling transparent bridging is recommended only on NNI ports that are known to and controlled by the network administrator.

The following command example configures the vendor TPID for NNI port 2/1 to 0x88a8 and enables support for Spanning Tree legacy BPDU:

```
-> ethernet-service nni 2/1 tpid 88a8 stp legacy-bpdu enable
```

Consider the following when configuring NNI port parameter values:

- A mismatch of TPID values on NNI ports that are connected together is not supported; VLAN Stacking will not work between switches using different NNI TPID values.
- Enable legacy BPDU support only on VLAN Stacking network ports that are connected to legacy BPDU switches. Enabling legacy BPDU between AOS switches may cause flooding or an unstable network.
- If legacy BPDU is enabled on a network port while at same time BPDU flooding is enabled on user ports, make sure that tagged customer BPDUs are not interpreted by intermediate switches in the provider network.
- If the peer switch connected to the VLAN Stacking network port supports the Provider MAC address (i.e., STP, 802.1ad/D6.0 MAC), then enabling legacy BPDU support is not required on the network port. Refer to the following table to determine the type of STP or GVRP MAC used:

STP	
Customer MAC	{0x01, 0x80, 0xc2, 0x00, 0x00, 0x00}
Provider MAC address (802.1ad/D6.0)	{0x01, 0x80, 0xc2, 0x00, 0x00, 0x08}
Provider MAC address (Legacy MAC)	{0x01, 0x80, 0xc2, 0x00, 0x00, 0x00}
GVRP	
Customer MAC address	{0x01, 0x80, 0xc2, 0x00, 0x00, 0x21}
Provider MAC address	{0x01, 0x80, 0xc2, 0x00, 0x00, 0x0D}

- GVRP legacy BPDU are supported only on network ports that already have GVRP enabled for the port.
- STP legacy BPDU are supported only when the flat Spanning Tree mode is active on the switch.

Use the `show ethernet-service nni` command to display the NNI port configuration for the switch.

Configuring a VLAN Stacking Service Access Point

The `ethernet-service sap` command is used to configure a VLAN Stacking service access point (SAP). An SAP is assigned an ID number at the time it is configured. This ID number is then associated with the following VLAN Stacking components:

- **User Network Interface (UNI) ports.** See [“Configuring VLAN Stacking User Ports” on page 9-16.](#)
- **Customer VLANs (CVLANs).** See [“Configuring the Type of Customer Traffic to Tunnel” on page 9-17.](#)
- **SAP profile.** Each SAP is associated with a single profile. This profile contains attributes that are used to define traffic engineering parameters applied to traffic ingressing on UNI ports that are associated with the SAP. See [“Configuring a Service Access Point Profile” on page 9-18.](#)

The above components are all configured separately using different VLAN Stacking commands. The **ethernet-service sap** command is for creating a SAP ID and associating the ID with a VLAN Stacking service. For example, the following command creates SAP 20 and associates it with Video-Service:

```
-> ethernet-service sap 20 service-name Video-Service
```

To delete a VLAN Stacking SAP from the switch configuration, use the **no** form of the **ethernet-service sap** command. For example, the following command deletes SAP 20:

```
-> no ethernet-service sap 20
```

Note that when the SAP is deleted, all UNI port, CVLAN, and profile associations are automatically dropped. It is not necessary to remove these items before deleting the SAP.

A VLAN Stacking SAP basically identifies the location where customer traffic enters the provider network edge, the type of customer traffic to service, parameters to apply to the traffic, and the service that will process the traffic for tunneling through the provider network.

Consider the following when configuring a VLAN Stacking SAP:

- A SAP is assigned to only one service, but a service can have multiple SAPs. So, a single service can process and tunnel traffic for multiple UNI ports and customers.
- Associating multiple UNI ports to one SAP is allowed.
- A default SAP profile is associated with the SAP at the time the SAP is created. This profile contains the following default attribute values:

Ingress bandwidth sharing	shared
Ingress bandwidth maximum	0
CLAN tag	preserve (double-tag)
Priority mapping	fixed 0

The above default attribute values are applied to customer traffic associated with the SAP. Only one profile is assigned to each SAP; however, it is possible to use the same profile for multiple SAPs.

- To use different profile attribute values, create a new profile and associate it with the SAP. See [“Configuring a Service Access Point Profile” on page 9-18](#). Each time a profile is assigned to a SAP, the existing profile is overwritten with the new one.

Use the **show ethernet-service sap** command to display the SAPs configured for the switch. Use the **show ethernet-service** command to display a list of VLAN Stacking services and the SAPs associated with each service.

Configuring VLAN Stacking User Ports

The **ethernet-service sap uni** command is used to configure a switch port or a link aggregate as a VLAN Stacking User Network Interface (UNI) and associate the UNI with a VLAN Stacking service access point (SAP). For example, the following command configures port 1/1 as an UNI port and associates 1/1 with SAP 20:

```
-> ethernet-service sap 20 uni 1/1
```

A UNI port is a customer-facing port on which traffic enters the VLAN Stacking service. When the port is associated with a service access point, the port is automatically defined as a UNI port and the default VLAN for the port is changed to a VLAN that is reserved for the VLAN Stacking application.

To delete a UNI port association with a VLAN Stacking SAP, use the **no** form of the **ethernet-service sap uni** command. For example, the following command deletes the association between UNI 1/1 and SAP 20:

```
-> ethernet-service sap 20 no uni 1/1
```

Note that when the last SAP association for the port is deleted, the port automatically reverts back to a conventional switch port and is no longer VLAN Stacking capable.

Consider the following when configuring VLAN Stacking UNI ports:

- All customer traffic received on the UNI port is dropped until customer VLANs (CVLAN) are associated with the port. See [“Configuring the Type of Customer Traffic to Tunnel” on page 9-17](#).
- If the SAP ID specified with this command is associated with an IPMVLAN, the SAP profile must specify CVLAN translation. In addition, multicast traffic is not associated with the IPMVLAN until the UNI port is associated with the IPMVLAN as a receiver port. For more information, see the [“Configuring IP Multicast VLANs”](#) chapter in this guide.
- A default UNI profile is assigned to the port at the time the port is configured. This profile defines how control frames received on the UNI ports are processed. By default, GVRP and Spanning Tree frames are tunneled. All other protocol control frames are dropped.
- To use different profile attribute values, create a new profile and associate it with the UNI port. See [“Configuring a UNI Profile” on page 9-20](#). Each time a profile is assigned to a UNI, the existing profile is overwritten with the new one.

Use the **show ethernet-service uni** command to display a list of UNI ports and the profile association for each port.

Configuring the Type of Customer Traffic to Tunnel

The **ethernet-service sap cvlan** command is used to associate customer traffic with a VLAN Stacking service access point (SAP). This identifies the type of customer traffic received on the SAP UNI ports that the service will process and tunnel through the SVLAN configured for the service. For example, the following command specifies that traffic tagged with customer VLAN (CVLAN) 500 is allowed on UNI ports associated with SAP 20:

```
-> ethernet-service sap 20 cvlan 500
```

In this example, customer frames tagged with VLAN ID 500 that are received on SAP 20 UNI ports are processed by the service to which SAP 20 is associated. This includes applying profile attributes associated with SAP 20 to the qualifying customer frames. If no other customer traffic is specified for SAP 20, all other frames received on SAP 20 UNI ports are dropped.

In addition to specifying one or more CVLANs, it is also possible to specify the following parameters when using the **ethernet-service sap cvlan** command:

- **all**—Specifies that all untagged and tagged frames are accepted on the UNI ports. If this parameter is combined with a CVLAN ID and bandwidth sharing and rate limiting are enabled for the SAP profile, then frames tagged with the CVLAN ID are given a higher bandwidth priority than all other frames received on the port.
- **untagged**—Specifies that only untagged frames are accepted on the UNI ports. If this parameter is combined with a CVLAN ID, then all untagged frames plus frames tagged with the CVLAN ID are accepted on the UNI ports.

For example, the following command specifies that all untagged frames and frames tagged with CVLAN ID 500 is accepted on UNI ports associated with SAP 20:

```
-> ethernet-service sap 20 cvlan 500 untagged
```

Use the **no** form of the **ethernet-service sap cvlan** command to delete an association between customer traffic and a VLAN Stacking SAP. For example, the following command deletes the association between CVLAN 500 and SAP 20:

```
-> ethernet-service sap 20 no cvlan 500
```

Note that when the last customer traffic association is deleted from a SAP, the SAP itself is not automatically deleted. No traffic is accepted or processed by a SAP in this state, but the SAP ID is still known to the switch.

Consider the following when configuring the type of customer traffic to tunnel:

- If no customer traffic is associated with a VLAN Stacking SAP, then the SAP does not process any traffic for the service.
- Only one **all** or **untagged** designation is allowed for any given SAP; specifying both for the same SAP is not allowed.
- Only one **untagged** designation is allowed per UNI port, even if the UNI port is associated with multiple SAPs.
- Only one **all** designation is allowed per UNI port, even if the UNI port is associated with multiple SAPs.
- Associating customer traffic with a service using an IP Multicast VLAN (IPMVLAN) is not allowed.

Use the **show ethernet-service** command to display the type of customer traffic associated with each SAP configured for the switch

Configuring a Service Access Point Profile

The **ethernet-service sap-profile** command is used to create a VLAN Stacking service access point (SAP) profile. The following command parameters define the traffic engineering attributes that are applied to customer traffic that is accepted on UNI ports associated with the SAP profile:

Profile Attribute	Command Parameters	Description
Ingress bandwidth sharing	shared not shared	Whether or not ingress bandwidth is shared across UNI ports and CVLANs.
Ingress rate limiting	ingress-bandwidth	The rate at which customer frames ingress on UNI ports.
Tri-Color Marking (TCM)	cir cbs pir pbs	Configures committed and peak information rate and burst size values to rate limit frames ingressing on UNI ports. See Chapter 34, “Configuring QoS,” for details about configuring TCM.
Double-tag or translate	cvlan preserve translate	Determines if a customer frame is tagged with the SVLAN ID (double-tag) or the CVLAN ID is changed to the SVLAN ID (translate) when the frame is encapsulated for tunneling. Double-tag is used by default.

Profile Attribute	Command Parameters	Description
Priority mapping	map-inner-to-outer-p map-dscp-to-outer-p fixed	Determines if the CVLAN (inner tag) 802.1p or DSCP value is mapped to the SVLAN (outer tag) 802.1p value or if a fixed priority value is used for the SVLAN 802.1p value. Priority mapping is set to a fixed rate of zero by default.

A default profile, named “default-sap-profile”, is automatically assigned to the SAP at the time the SAP is created (see [“Configuring a VLAN Stacking Service Access Point” on page 9-15](#)). It is only necessary to create a new profile to specify different attribute values if the default profile values (see above) are not sufficient.

The following command provides an example of creating a new SAP profile to specify a different method for mapping the SVLAN priority value:

```
-> ethernet-service sap-profile map_pbit priority map-inner-to-outer-p
```

In this example the **map_pbit** profile specifies priority mapping of the CVLAN inner tag 802.1p value to the SVLAN outer tag value. The other attributes in this profile are set to their default values.

To delete a SAP profile, use the **no** form of the **ethernet-service sap-profile** command. For example, the following command deletes the **map_pbit** profile:

```
-> no ethernet-service sap-profile map_pbit
```

Use the **show ethernet-service sap-profile** command to view a list of profiles that are already configured for the switch. This command also displays the attribute values for each profile.

Associating a Profile with a Service Access Point

After a profile is created, it is then necessary to associate the profile with a VLAN Stacking SAP. When this is done, the current profile associated with a SAP is replaced with the new profile.

The **ethernet-service sap sap-profile** command is used to associate a new profile with a VLAN Stacking SAP. For example, the following command associates the `map_pbit` profile to SAP 20:

```
-> ethernet-service sap 20 sap-profile map_pbit
```

Note the following when associating a profile with a VLAN Stacking SAP:

- To change the profile associated with the SAP back to the default profile, specify “default-sap-profile” for the profile name. For example:

```
-> ethernet-service sap 20 sap-profile default-sap-profile
```

- If the SAP ID specified with this command is associated with an IPMVLAN, the profile associated with the SAP ID must specify CVLAN tag translation. Double tagging is not supported with IPMVLAN SAPs that are also associated with a UNI port.

Use the **show ethernet-service sap** command to display the SAP configuration, which includes the profile association for each SAP.

Configuring a UNI Profile

The **ethernet-service uni-profile** command is used to create a VLAN Stacking UNI port profile. The UNI profile determines how control frames ingressing on UNI ports are processed. For example, the following command creates a UNI profile to specify that VLAN Stacking should discard GVRP frames:

```
-> ethernet-service uni-profile discard-gvrp l2-protocol gvrp discard
```

A default UNI profile, named “default-uni-profile”, is automatically associated with a UNI port. The default UNI profile specifies how control frames ingressing on the UNI port.

To delete a UNI profile, use the **no** form of the **ethernet-service uni-profile** command. For example, the following command deletes the **discard-gvrp** profile:

```
-> no ethernet-service uni-profile discard-gvrp
```

Use the **show ethernet-service uni-profile** command to view a list of profiles that are already configured for the switch. This command also displays the attribute values for each profile.

Note. The VLAN Stacking provider edge (PE) switch will not tunnel GVRP frames unless the GVRP feature and/or GVRP transparent switching functionality is enabled on the PE switch. This is true even if GVRP processing is enabled for the VLAN Stacking port.

Configuring Destination MAC Address

The **ethernet-service uni-profile** command can also be used to configure the destination MAC address of L2 protocol control packets as they are sent through the provider network. Each protocol has a default tunnel MAC address or a user specified destination MAC address can be configured. For example the following command configures the VRP protocol to use the configured tunnel MAC address instead of the default protocol destination MAC address:

```
-> ethernet-service uni-profile uni_1 l2-protocol vrp mac-tunnel
```

Associating UNI Profiles with UNI Ports

After a UNI profile is created, it is then necessary to associate the profile with a UNI port or a UNI link aggregate. When this is done, the current profile associated with the port is replaced with the new profile.

The **ethernet-service uni uni-profile** command is used to associate a new profile with a UNI port. For example, the following command associates the discard-gvrp profile to UNI port 1/1:

```
-> ethernet-service uni 1/1 uni-profile discard-gvrp
```

To change the profile associated with the UNI port back to the default profile, specify “default-uni-profile” for the profile name. For example:

```
-> ethernet-service uni 1/1 uni-profile default-uni-profile
```

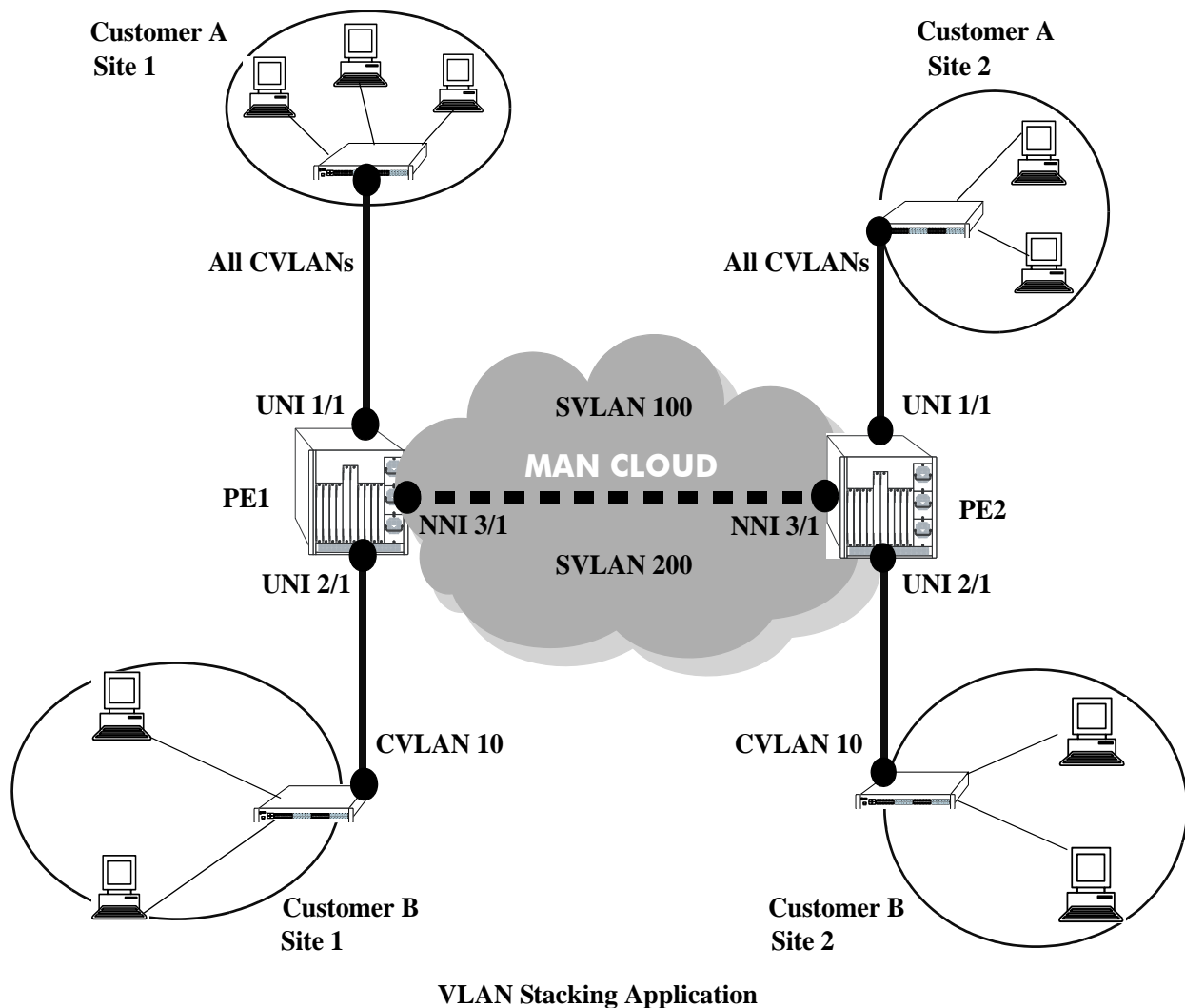
Use the **show ethernet-service uni** command to display the profile associations for each UNI port.

VLAN Stacking Application Examples

The VLAN Stacking feature provides the ability to transparently connect multiple customer sites over a single shared service provider network. This section demonstrates this ability by providing a sample VLAN Stacking configuration that tunnels customer VLANs (CVLAN) inside a service provider VLAN (SVLAN) so that customer traffic is transparently bridged through a Metropolitan Area Network (MAN).

The illustration below shows the sample VLAN Stacking configuration described in this section. In this configuration, the provider edge bridges will encapsulate Customer A traffic (all CVLANs) into SVLAN 100 and Customer B traffic (CVLAN 10 only) into SVLAN 200. In addition, the CVLAN 10 inner tag priority bit value is mapped to the SVLAN out tag priority value. The customer traffic is then transparently bridged across the MAN network and sent out to the destined customer site.

Double-tagging is the encapsulation method used in this application example. This method consists of appending the SVLAN tag to customer packets ingressing on provider edge UNI ports so that the traffic is bridged through the provider network SVLAN. The SVLAN tag is then stripped off of customer packets egressing on provider edge UNI ports before the packets are delivered to their destination customer site.



VLAN Stacking Configuration Example

This section provides a tutorial for configuring the sample application, as illustrated on [page 9-22](#), using VLAN Stacking Ethernet services. This tutorial assumes that both provider edge switches (PE1 and PE2) are operating in the VLAN Stacking service mode.

1 Configure SVLAN 100 and SVLAN 200 on PE1 *and* PE2 switches using the **ethernet-service** command.

```
-> ethernet-service svlan 100
-> ethernet-service svlan 200
```

2 Configure two VLAN Stacking services on PE1 *and* PE2 using the **ethernet-service service-name** command. Configure one service with the name “CustomerA” and the other service with the name “Customer B”. Assign “CustomerA” service to SVLAN 100 and “CustomerB” service to SVLAN 200.

```
-> ethernet-service service-name CustomerA svlan 100
-> ethernet-service service-name CustomerB svlan 200
```

3 Configure port 3/1 on PE1 *and* PE2 as VLAN Stacking NNI ports using the **ethernet-service svlan nni** command. Associate each port with both SVLAN 100 and SVLAN 200.

```
-> ethernet-service svlan 100 nni 3/1
-> ethernet-service svlan 200 nni 3/1
```

4 Configure a VLAN Stacking SAP with ID 20 on PE1 *and* PE2 using the **ethernet-service sap**. Associate the SAP with the “CustomerA” service.

```
-> ethernet-service sap 20 service-name CustomerA
```

5 Configure a VLAN Stacking SAP with ID 30 on PE1 *and* PE2 using the **ethernet-service sap** command. Associate the SAP with the “CustomerB” service.

```
-> ethernet-service sap 30 service-name CustomerB
```

6 Configure port 1/1 on PE1 *and* PE2 as a VLAN Stacking UNI port and associate 1/1 with SAP 20 using the **ethernet-service sap uni** command.

```
-> ethernet-service sap 20 uni 1/1
```

7 Configure port 2/1 on PE1 *and* PE2 as a VLAN Stacking UNI port and associate 2/1 with SAP 30 using the **ethernet-service sap uni** command.

```
-> ethernet-service sap 30 uni 2/1
```

8 Configure SAP 20 on PE1 *and* PE2 to accept all customer traffic on UNI port 1/1 using the **ethernet-service sap cvlan** command.

```
-> ethernet-service sap 20 cvlan all
```

9 Configure SAP 30 on PE1 *and* PE2 to accept only customer traffic that is tagged with CVLAN 10 using the **ethernet-service sap cvlan** command.

```
-> ethernet-service sap 30 cvlan 10
```

10 Create a SAP profile on PE1 and PE2 that will map the inner CVLAN tag 802.1p value to the outer SVLAN tag using the **ethernet-service sap-profile** command.

```
-> ethernet-service sap-profile map_pbit priority map-inner-to-outer-p
```

11 Associate the “map_pbit” profile to SAP 30 using the **ethernet-service sap sap-profile** command. This profile will only apply to Customer B traffic, so it is not necessary to associate the profile with SAP 20.

```
-> ethernet-service sap 30 sap-profile map_pbit
```

12 Verify the VLAN Stacking service configuration using the **show ethernet-service** command.

```
-> show ethernet-service
```

```
Service Name : CustomerA
  SVLAN      : 100
  NNI(s)     : 3/1
  SAP Id     : 20
    UNIs      : 1/1
    CVLAN(s)  : all
  sap-profile : default-sap-profile
```

```
Service Name : CustomerB
  SVLAN      : 200
  NNI(s)     : 3/1
  SAP Id     : 10
    UNIs      : 2/1
    CVLAN(s)  : 10
  sap-profile : map_pbit
```

The following is an example of what the sample configuration commands look like entered sequentially on the command line of the provider edge switches:

```
-> ethernet-service svlan 100
-> ethernet-service service-name CustomerA svlan 100
-> ethernet-service svlan 100 nni 3/1
-> ethernet-service sap 20 service-name CustomerA
-> ethernet-service sap 20 uni 1/1
-> ethernet-service sap 20 cvlan all

-> ethernet-service svlan 200
-> ethernet-service service-name CustomerB svlan 200
-> ethernet-service svlan 200 nni 3/1
-> ethernet-service sap 30 service-name CustomerB
-> ethernet-service sap 30 uni 2/1
-> ethernet-service sap 30 cvlan 10
-> ethernet-service sap-profile map_pbit priority map-inner-to-outer-p
-> ethernet-service sap 30 sap-profile map_pbit
```


Wire-Speed Ethernet Loopback Test

A wire-speed Ethernet loopback test function is available to perform In-Service and Out-of-Service throughput testing during initial turn-up or on-the-fly in an active network. The loopback tests can be used to validate the configured Service Level Agreements (SLAs) and QoS parameters that are associated with a service or a flow.

The loopback test capability provided allows the use of an external test head to send traffic at wire-rate speed to a specific switch port which then loops the traffic back to the test head. The test head measures and collects statistics on frame loss, delay, and latency of the loopback traffic.

There are two types of loopback tests supported with this implementation: inward loopback and outward loopback. The inward test loops back test head frames ingressing on a given port. The outward test loops back test head frames egressing on a given port.

Configuring an Ethernet Loopback Test

The type of loopback test performed is determined by a user-configured test profile that specifies the following information:

- The name of the test profile.
- A unique source MAC address for the test frames. In this case, the MAC address of the device that will generate the test frames is used.
- A unique destination MAC address for the test frames. For an inward test, using the base MAC address of the destination switch is recommended. For an outward test, use the base MAC address of customer premises equipment (CPE) or the MAC address of the egress port on the provider edge (PE) switch.
- The VLAN ID on which the test frames are forwarded (if the frame is double-tagged, this is the VLAN ID of the outer tag).
- The switch port (for example, the UNI or NNI port) that will perform the egress or ingress loopback operation for the test.
- The type of test to run (outward or inward loopback).

The **loopback-test** command is used to define the test profile and is also the same command that is used to enable or disable the actual loopback operation. For example, the following command creates an inward loopback test profile:

```
-> loopback-test PE1-inward-UNI source-mac 00:00:00:dd:aa:01 destination-mac
00:00:00:cc:aa:bb vlan 1001 loopback-port 1/1 type inward
```

The following commands enable and disable the **PE1-inward-UNI** profile attributes for the switch:

```
-> loopback-test PE1-inward-UNI enable
-> loopback-test PE1-inward-UNI disable
```

Use the **show loopback-test** command to display the loopback test profile configuration.

Consider the following guidelines when configuring an Ethernet loopback test:

- Up to eight profiles are configurable per switch.
- Test frames must have an Ethertype of 0x800 (IP frames).
- Only Layer 2 loopback tests are supported: test frames are not routed. The loopback operation will only swap the source and destination MAC address of bridged test frames.
- The switch creates a static MAC address entry for the egress port when the outward loopback profile is applied on that port. The static address created is the destination MAC address specified in the profile. If the switch receives a non-test frame that contains the same MAC address, both the test and non-test frames are filtered even if they were received on different ports.
- Each loopback test is associated with one VLAN; using multiple VLANs is not supported.
- Once a port is designated as the loopback port for a test, that port is no longer available for use by other switch applications.
- Ports used for an outward loopback operation go “out-of-service” and will no longer carry customer traffic. The port does remain active, however, for test frame traffic.
- Ports used for an inward loopback operation remain “in-service”. Test frame traffic is mixed in with customer frame traffic.
- If the MAC addresses specified in the loopback test profile is an actual network address (for example, 02:da:95:e1:22:10, not aa:aa:aa:aa:aa:aa), flush the MAC address table for the switch when loopback testing is finished.

The following sections provide more information about using and configuring both types of Ethernet loopback tests.

Outward (Egress) Loopback Test

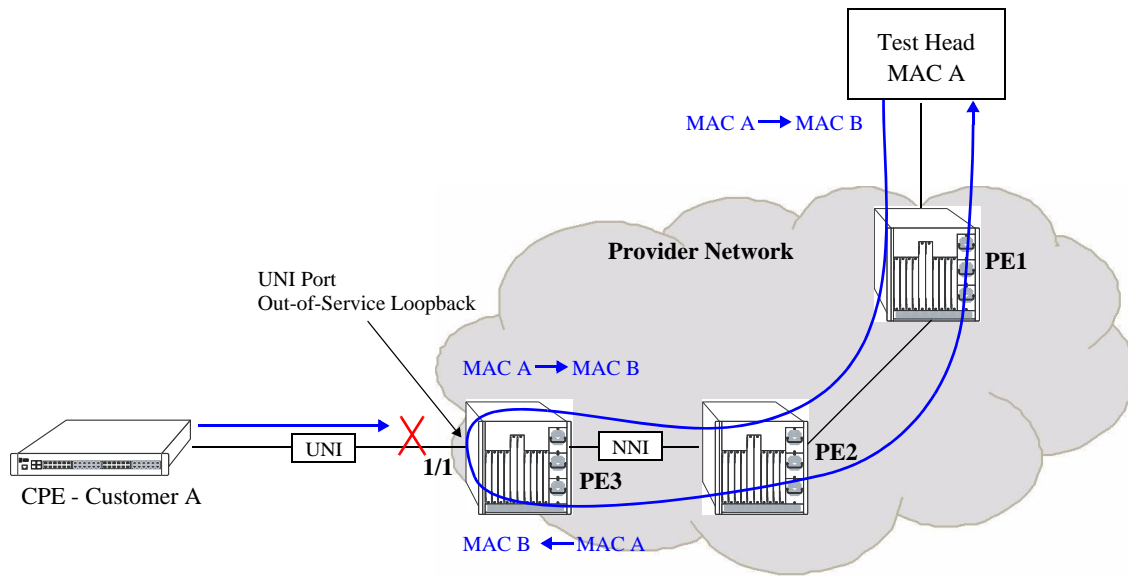
An outward loopback test loops back test frames egressing on a specific port. The source and destination MAC addresses of the test frames are swapped and the frames are then redirected back to the port on which they were initially received and learned (the redirect port). The redirect port is not configured as part of the test profile; the source learning function determines which port to use based on the known source MAC and VLAN of the test frames.

This type of test renders the loopback port “out-of-service”, which means the port is no longer available to forward customer traffic. Although customer frames are dropped, the port does remain in an up state and is active for looping back test frames.

Typically, an outward loopback operation is configured and performed on a UNI port. Test frames egressing on the UNI port are looped back on to the UNI port where the frames are processed as if they were sent from a customer site. As a result, the attributes of the Ethernet Services SAP profile associated with the UNI port are applied to the test frames before they are sent back to the redirect port.

The following illustration shows an example of an outward loopback test operation in which the loopback operation is configured on a UNI port of a provider edge switch.

Note. Conducting an outward loopback test disrupts the flow of customer traffic on the loopback port and may cause network reachability problems.



Outward (Egress) Loopback Test Example

In this outward loopback test example:

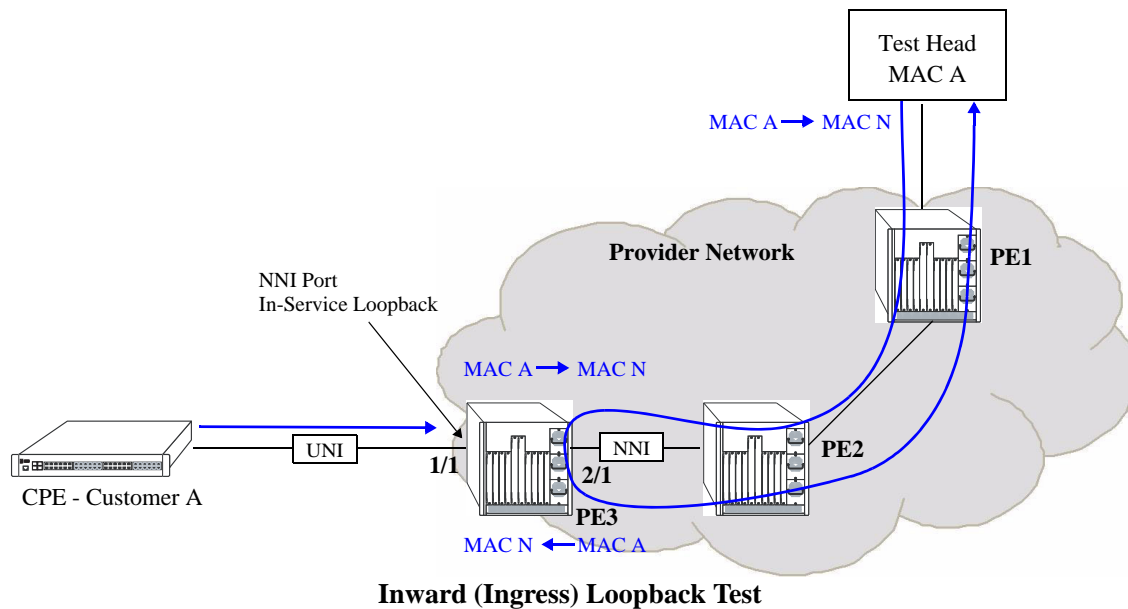
- An outward loopback test profile is configured and enabled for UNI port 1/1 on PE3. The source MAC address for the profile is that of the test head (MAC A); the destination MAC address is a static MAC address configured for the UNI port (MAC B).
- UNI port 1/1 on PE3 is out of service for customer traffic.
- The test head transmits frames with source MAC A and destination MAC B.
- When the test frames reach UNI port 1/1 on PE3, the egress loopback operation is triggered on that port. MAC A and B are swapped in each test frame as the frames are looped back on to the egress port.
- Once the egress loopback operation is complete, the frames are then sent to the redirect port and forwarded back to the test head.

Inward (Ingress) Loopback Test

An inward loopback test loops back test frames ingressing on a specific port. The source and destination MAC addresses of the test frames are swapped and the frames are then redirected back to the same port. In other words, the ingress port is both the loopback and redirect port.

This type of test allows the ingress loopback port to remain “in-service” for customer traffic. As a result, customer frames and test frames are both serviced on the loopback port; there is no disruption to customer traffic.

The following illustration shows an example of an inward loopback test operation in which the loopback operation is configured on a NNI port of a provider edge switch.



In this inward loopback example:

- An inward loopback test profile is configured and enabled for NNI port 2/1 on PE3. The source MAC address for the profile is that of the test head (MAC A); the destination MAC address is the switch base MAC address for PE3 (MAC N).
- NNI port 2/1 on PE3 is in-service for customer traffic and test frames.
- The test head transmits frames with source MAC A and destination MAC N.
- When the test frames reach NNI port 2/1 on PE3, the ingress loopback operation is triggered on that port. MAC A and N are swapped in each test frame as the frames are looped back onto the ingress port.
- Once the ingress loopback operation is complete and because the NNI port is also the redirect port in this case, the frames are forwarded back to the test head.

Verifying the VLAN Stacking Configuration

You can use CLI **show** commands to display the current configuration and statistics of service-based VLAN Stacking on a switch. These commands include the following:

show ethernet-service mode	Displays the active VLAN Stacking mode for the switch.
show ethernet-service vlan	Displays the SVLAN configuration for the switch.
show ethernet-service	Displays the VLAN Stacking service configuration for the switch.
show ethernet-service sap	Displays the VLAN Stacking service access point (SAP) configuration for the switch.
show ethernet-service port	Displays configuration information for VLAN Stacking ports.
show ethernet-service nni	Displays configuration information for NNI port parameters.
show ethernet-service uni	Displays profile associations for UNI ports.
show ethernet-service uni-profile	Displays UNI profile attribute values.
show ethernet-service sap-profile	Displays SAP profile attribute values.
show ethernet-service statistics	Displays Tri-Color Marking (TCM) results by showing the number of packets marked green, yellow, and red.

For more information about the resulting displays from these commands, see the *OmniSwitch CLI Reference Guide*. An example of the output for the **show ethernet-service** command is also given in [“Quick Steps for Configuring VLAN Stacking”](#) on page 9-9.

10 Using 802.1Q 2005 Multiple Spanning Tree

The Alcatel-Lucent Multiple Spanning Tree (MST) implementation provides support for the Multiple Spanning Tree Protocol (MSTP) as defined in the IEEE 802.1Q 2005 standard. In addition to the 802.1D Spanning Tree Algorithm and Protocol (STP) and the 802.1w Rapid Spanning Tree Algorithm and Protocol (RSTP), MSTP also ensures that there is always only one data path between any two switches for a given Spanning Tree instance to prevent network loops.

MSTP is an enhancement to the 802.1Q Common Spanning Tree (CST), which is provided when an Alcatel-Lucent switch is running in the flat Spanning Tree operating mode. The flat mode applies a single spanning tree instance across all VLAN port connections on a switch. MSTP allows the configuration of Multiple Spanning Tree Instances (MSTIs) in addition to the CST instance. Each MSTI is mapped to a set of VLANs. As a result, flat mode can support the forwarding of VLAN traffic over separate data paths.

In addition to MSTP support, the STP and RSTP are still available in either the flat or 1x1 mode. However, if using STP or RSTP in the flat mode, the single Spanning Tree instance per switch algorithm applies.

In This Chapter

This chapter describes MST in general and how MSTP works on the switch. It provides information about configuring MSTP through the Command Line Interface (CLI). For more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*. For more information about Spanning Tree configuration commands as they apply to all supported protocols (STP, RSTP, and MSTP), see [Chapter 11, “Configuring Spanning Tree Parameters.”](#)

The following topics are included in this chapter as they relate to the Alcatel-Lucent implementation of the MSTP standard:

- [“MST General Overview” on page 10-4.](#)
- [“MST Configuration Overview” on page 10-10.](#)
- [“Using Spanning Tree Configuration Commands” on page 10-10.](#)
- [“MST Interoperability and Migration” on page 10-12.](#)
- [“Quick Steps for Configuring an MST Region” on page 10-14.](#)
- [“Quick Steps for Configuring MSTIs” on page 10-16.](#)
- [“Verifying the MST Configuration” on page 10-19.](#)

Spanning Tree Specifications

IEEE Standards supported	802.1D— <i>Media Access Control (MAC) Bridges</i> 802.1w— <i>Rapid Reconfiguration (802.1D Amendment 2)</i> 802.1Q 2005— <i>Virtual Bridged Local Area Networks</i>
Spanning Tree Protocols supported	802.1D Standard Spanning Tree Algorithm and Protocol (STP) 802.1w Rapid Spanning Tree Algorithm and Protocol (RSTP) Multiple Spanning Tree Algorithm and Protocol (MSTP)
Platforms Supported	OmniSwitch 6250 Series
Spanning Tree Operating Modes supported	Flat mode - one spanning tree instance per switch 1x1 mode - one spanning tree instance per VLAN
Spanning Tree port eligibility	Fixed ports (non-mobile) 802.1Q tagged ports Link aggregate of ports
Maximum 1x1 Spanning Tree instances per switch	252
Maximum flat mode Multiple Spanning Tree Instances (MSTI) per switch	16 MSTI, in addition to the Common and Internal Spanning Tree instance (also referred to as MSTI 0).
Number of Ring Rapid Spanning Tree (RRSTP) rings supported	8
CLI Command Prefix Recognition	All Spanning Tree commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

Spanning Tree Bridge Parameter Defaults

Parameter Description	Command	Default
Spanning Tree operating mode	bridge mode	1x1 (a separate Spanning Tree instance for each VLAN)
Spanning Tree protocol	bridge protocol	RSTP (802.1w)
Priority value for a Multiple Spanning Tree Instance (MSTI).	bridge msti priority	32768
Hello time interval between each BPDU transmission.	bridge hello time	2 seconds
Maximum aging time allowed for Spanning Tree information learned from the network.	bridge max age	20 seconds
Spanning Tree port state transition time.	bridge forward delay	15 seconds
BPDU switching status.	bridge bpdu-switching	Disabled
Path cost mode	bridge path cost mode	AUTO (16-bit in 1x1 mode, 32-bit in flat mode)

Parameter Description	Command	Default
Automatic VLAN Containment	bridge auto-vlan-containment	Disabled

Spanning Tree Port Parameter Defaults

Parameter Description	Command	Default
Spanning Tree port administrative state	bridge slot/port	Enabled
Port priority value for a Multiple Spanning Tree instance	bridge msti slot/port priority	7
Port path cost for a Multiple Spanning Tree instance	bridge msti slot/port path cost	0 (cost is based on port speed)
Port state management mode	bridge slot/port mode	Dynamic (Spanning Tree Algorithm determines port state)
Type of port connection	bridge slot/port connection	auto point to point

Multiple Spanning Tree Region Defaults

Although the following parameter values are specific to MSTP, they are configurable regardless of which mode (flat or 1x1) or protocol is active on the switch.

Parameter Description	Command	Default
The Multiple Spanning Tree region name	bridge mst region name	blank
The revision level for the Multiple Spanning Tree region	bridge mst region revision level	0
The maximum number of hops authorized for the region	bridge mst region max hops	20
The number of Multiple Spanning Tree instances	bridge msti	1 (flat mode instance)
The VLAN to Multiple Spanning Tree instance mapping.	bridge msti vlan	All VLANs are mapped to the Common Internal Spanning Tree (CIST) instance

MST General Overview

The Multiple Spanning Tree (MST) feature allows for the mapping of one or more VLANs to a single Spanning Tree instance, referred to as a Multiple Spanning Tree Instance (MSTI), when the switch is running in the flat Spanning Tree mode. MST uses the Multiple Spanning Tree Algorithm and Protocol (MSTP) to define the Spanning Tree path for each MSTI. In addition, MSTP provides the ability to group switches into MST Regions. An MST Region appears as a single, flat Spanning Tree instance to switches outside the region.

This section provides an overview of the MST feature that includes the following topics:

- [“How MSTP Works” on page 10-4.](#)
- [“Comparing MSTP with STP and RSTP” on page 10-7.](#)
- [“What is a Multiple Spanning Tree Instance \(MSTI\)” on page 10-7.](#)
- [“What is a Multiple Spanning Tree Region” on page 10-8.](#)
- [“What is the Internal Spanning Tree \(IST\) Instance” on page 10-9.](#)
- [“What is the Common and Internal Spanning Tree Instance” on page 10-9.](#)

How MSTP Works

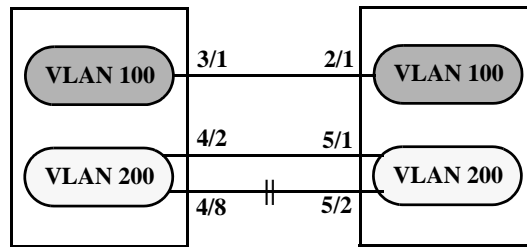
MSTP, as defined in the IEEE 802.1Q 2005 standard, is an enhancement to the IEEE 802.1Q Common Spanning Tree (CST). The CST is a single spanning tree that uses 802.1D (STP) or 802.1w (RSTP) to provide a loop-free network topology.

The Alcatel-Lucent flat spanning tree mode applies a single CST instance on a per switch basis. The 1x1 mode is an Alcatel-Lucent proprietary implementation that applies a single spanning tree instance on a per VLAN basis. MSTP is only supported in the flat mode and allows for the configuration of additional spanning tree instances instead of just the one CST.

On Alcatel-Lucent MSTP flat mode switches, the CST is represented by the Common and Internal Spanning Tree (CIST) instance 0 and exists on all switches. Up to 17 instances, including the CIST, are supported. Each additional instance created is referred to as a Multiple Spanning Tree Instance (MSTI). An MSTI represents a configurable association between a single Spanning Tree instance and a set of VLANs.

Note that although MSTP provides the ability to define MSTIs while running in the flat mode, port state and role computations are still automatically calculated by the CST algorithm across all MSTIs. However, it is possible to configure the priority and/or path cost of a port for a particular MSTI so that a port remains in a forwarding state for an MSTI instance, even if it is blocked as a result of automatic CST computations for other instances.

The following diagrams help to further explain how MSTP works by comparing how port states are determined on 1x1 STP/RSTP mode, flat mode STP/RSTP, and flat mode MSTP switches.



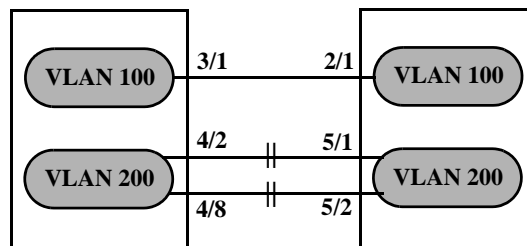
1x1 Mode STP/RSTP

In the above 1x1 mode example:

- Both switches are running in the 1x1 mode (one Spanning Tree instance per VLAN).
- VLAN 100 and VLAN 200 are each associated with their own Spanning Tree instance.
- The connection between 3/1 and 2/1 is left in a forwarding state because it is part of the VLAN 100 Spanning Tree instance and is the only connection for that instance.

Note that if additional switches containing a VLAN 100 were attached to the switches in this diagram, the 3/1 to 2/1 connection could also go into blocking if the VLAN 100 Spanning Tree instance determines it is necessary to avoid a network loop.

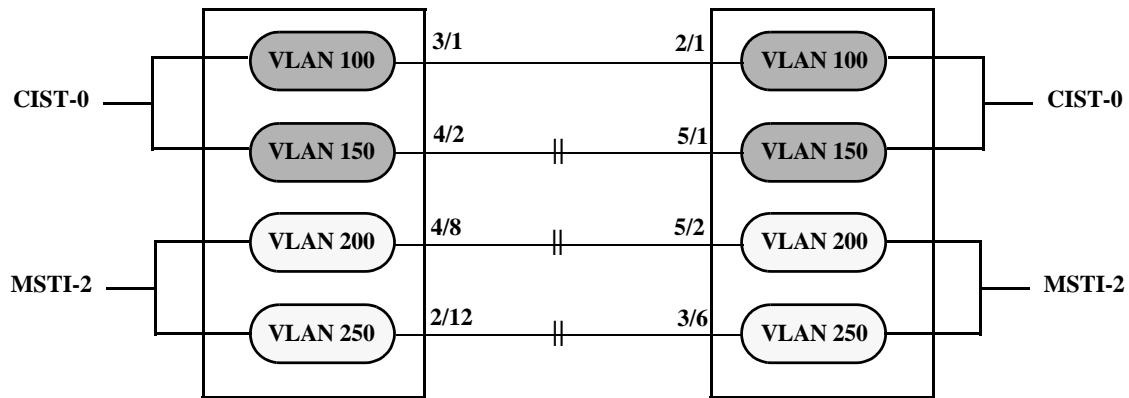
- The connections between 4/8 and 5/2 and 4/2 and 5/1 are seen as redundant because they are both controlled by the VLAN 200 Spanning Tree instance and connect to the same switches. The VLAN 200 Spanning Tree instance determines which connection provides the best data path and transitions the other connection to a blocking state.



Flat Mode STP/RSTP (802.1D/802.1w)

In the above flat mode STP/RSTP example:

- Both switches are running in the flat mode. As a result, a single flat mode Spanning Tree instance applies to the entire switch and compares port connections across VLANs to determine which connection provides the best data path.
- The connection between 3/1 and 2/1 is left forwarding because the flat mode instance determined that this connection provides the best data path between the two switches.
- The 4/8 to 5/2 connection and the 4/2 to 5/1 connection are considered redundant connections so they are both blocked in favor of the 3/1 to 2/1 connection.



Flat Mode MSTP

In the above flat mode MSTP example:

- Both switches are running in the flat mode and using MSTP.
- VLANs 100 and 150 are *not* associated with an MSTI. By default they are controlled by the CIST instance 0, which exists on every switch.
- VLANs 200 and 250 are associated with MSTI 2 so their traffic can traverse a path different from that determined by the CIST.
- Ports are blocked the same way they were blocked in the flat mode STP/RSTP example; all port connections are compared to each other across VLANs to determine which connection provides the best path.

However, because VLANs 200 and 250 are associated to MSTI 2, it is possible to change the port path cost for ports 2/12, 3/6, 4/8 and/or 5/2 so that they provide the best path for MSTI 2 VLANs, but do not carry CIST VLAN traffic or cause CIST ports to transition to a blocking state.

Another alternative is to assign all VLANs to an MSTI, leaving no VLANs controlled by the CIST. As a result, the CIST BPDU will only contain MSTI information.

See [“Quick Steps for Configuring MSTIs” on page 10-16](#) for more information about how to direct VLAN traffic over separate data paths using MSTP.

Comparing MSTP with STP and RSTP

Using MSTP has the following items in common with STP (802.1D) and RSTP (802.1w) protocols:

- Each protocol ensures one data path between any two switches within the network topology. This prevents network loops from occurring while at the same time allowing for redundant path configuration.
- Each protocol provides automatic reconfiguration of the network Spanning Tree topology in the event of a connection failure and/or when a switch is added to or removed from the network.
- All three protocols are supported in the flat Spanning Tree operating mode.
- The flat mode CST instance automatically determines port states and roles across VLAN port and MSTI associations. This is because the CST instance is active on all ports and only one BPDU is used to forward information for all MSTIs.
- MSTP is based on RSTP.

Using MSTP differs from STP and RSTP as follows:

- MSTP is only supported when the switch is running in the flat Spanning Tree mode. STP and RSTP are supported in both the 1x1 and flat modes.
- MSTP allows for the configuration of up to 16 Multiple Spanning Tree Instances (MSTI) in addition to the CST instance. Flat mode STP and RSTP protocols only use the single CST instance for the entire switch. See [“What is a Multiple Spanning Tree Instance \(MSTI\)” on page 10-7](#) for more information.
- MSTP applies a single Spanning Tree instance to an MSTI ID number that represents a set of VLANs; a one to many association. STP and RSTP in the flat mode apply one Spanning Tree instance to all VLANs; a one to all association. STP and RSTP in the 1x1 mode apply a single Spanning Tree instance to each existing VLAN; a one to one association.
- The port priority and path cost parameters are configurable for an individual MSTI that represents the VLAN associated with the port.
- The flat mode 802.1D or 802.1w CST is identified as instance 1. When using MSTP, the CST is identified as CIST (Common and Internal Spanning Tree) instance 0. See [“What is the Common and Internal Spanning Tree Instance” on page 10-9](#) for more information.
- MSTP allows the segmentation of switches within the network into MST regions. Each region is seen as a single virtual bridge to the rest of the network, even though multiple switches may belong to the one region. See [“What is a Multiple Spanning Tree Region” on page 10-8](#) for more information.
- MSTP has lower overhead than a 1x1 configuration. In 1x1 mode, because each VLAN is assigned a separate Spanning Tree instance, BPDUs are forwarded on the network for each VLAN. MSTP only forwards one BPDU for the CST that contains information for all configured MSTI on the switch.

What is a Multiple Spanning Tree Instance (MSTI)

An MSTI is a single Spanning Tree instance that represents a group of VLANs. Alcatel-Lucent switches support up to 16 MSTIs on one switch. This number is in addition to the Common and Internal Spanning Tree (CIST) instance 0, which is also known as MSTI 0. The CIST instance exists on every switch. By default, all VLANs not mapped to an MSTI are associated with the CIST instance. See [“What is the Common and Internal Spanning Tree Instance” on page 10-9](#) for more information.

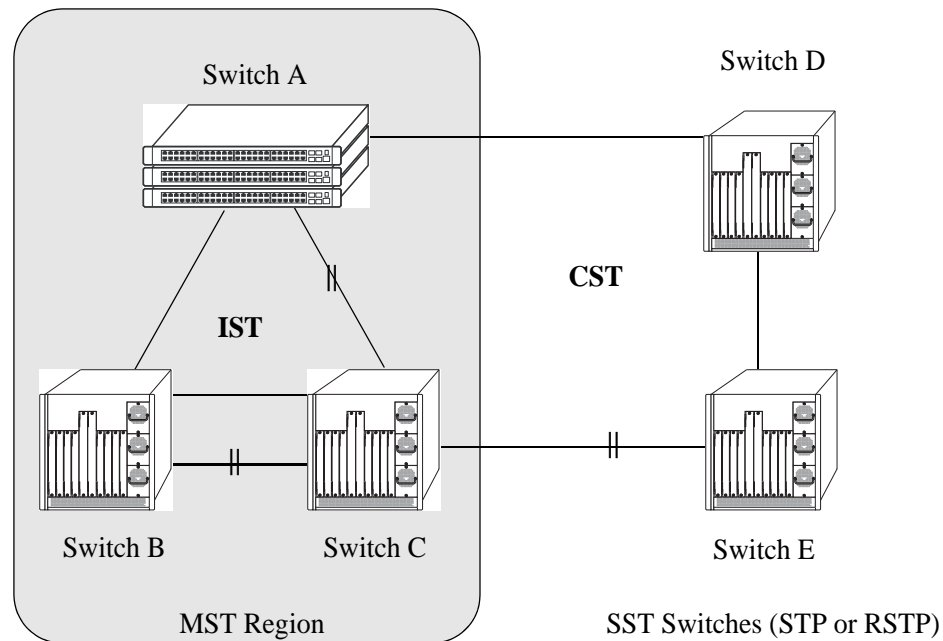
What is a Multiple Spanning Tree Region

A Multiple Spanning Tree region represents a group of MSTP switches. An MST region appears as a single, flat mode instance to switches outside the region. A switch can belong to only one region at a time. The region a switch belongs to is identified by the following configurable attributes, as defined by MSTP.

- **Region name**—An alphanumeric string up to 32 characters.
- **Region revision level**—A numerical value between 0 and 65535.
- **VLAN to MSTI table**—Generated when VLANs are associated with MSTIs. Identifies the VLAN to MSTI mapping for the switch.

Switches that share the same values for the configuration attributes described above belong to the same region. For example, in the diagram below:

- Switches A, B, and C all belong to the same region because they all are configured with the same region name, revision level, and have the same VLANs mapped to the same MSTI.
- The CST for the entire network sees Switches A, B, and C as one virtual bridge that is running a single Spanning Tree instance. As a result, CST blocks the path between Switch C and Switch E instead of blocking a path between the MST region switches to avoid a network loop.
- The paths between Switch A and Switch C and the redundant path between Switch B and Switch C were blocked as a result of the Internal Spanning Tree (IST) computations for the MST Region. See [“What is the Internal Spanning Tree \(IST\) Instance” on page 10-9](#) for more information.



In addition to the attributes described above, the MST maximum hops parameter defines the number of bridges authorized to propagate MST BPDU information. In essence, this value defines the size of the region in that once the maximum number of hops is reached, the BPDU is discarded.

The maximum number of hops for the region is not one of the attributes that defines membership in the region. See [“Quick Steps for Configuring an MST Region” on page 10-14](#) for a tutorial on how to configure MST region parameters.

What is the Common Spanning Tree

The Common Spanning Tree (CST) is the overall network Spanning Tree topology resulting from STP, RSTP, and/or MSTP calculations to provide a single data path through the network. CST provides connectivity between MST regions and other MST regions and/or Single Spanning Tree (SST) switches. For example, in the above diagram, CST calculations detected a network loop created by the connections between Switch D, Switch E, and the MST Region. As a result, one of the paths was blocked.

What is the Internal Spanning Tree (IST) Instance

The IST instance determines and maintains the CST topology between MST switches that belong to the same MST region. In other words, the IST is simply a CST that only applies to MST Region switches while at the same time representing the region as a single Spanning Tree bridge to the network CST.

As shown in the above diagram, the redundant path between Switch B and Switch C is blocked and the path between Switch A and Switch C is blocked. These blocking decisions were based on IST computations within the MST region. IST sends and receives BPDU to/from the network CST. MSTI within the region do not communicate with the network CST. As a result, the CST only sees the IST BPDU and treats the MST region as a single Spanning Tree bridge.

What is the Common and Internal Spanning Tree Instance

The Common and Internal Spanning Tree (CIST) instance is the Spanning Tree calculated by the MST region IST and the network CST. The CIST is represented by the single Spanning Tree flat mode instance that is available on all switches. By default, all VLANs are associated to the CIST until they are mapped to an MSTI.

When using STP (802.1D) or RSTP (802.1w), the CIST is also known as instance 1 or bridge 1. When using MSTP, the CIST is also known as instance 0 or MSTI 0.

Note that when MSTP is the active flat mode protocol, explicit Spanning Tree bridge commands are required to configure parameter values. Implicit commands are for configuring parameters when the STP or RSTP protocols are in use. See [“Using Spanning Tree Configuration Commands” on page 10-10](#) for more information.

MST Configuration Overview

The following general steps are required to set up a Multiple Spanning Tree (MST) configuration:

- **Select the flat Spanning Tree mode.** By default, each switch runs in the 1x1 mode. MSTP is only supported on a flat mode switch. See [“Understanding Spanning Tree Modes” on page 10-11](#) for more information.
- **Select the MSTP protocol.** By default, each switch uses the 802.1w protocol. Selecting MSTP activates the Multiple Spanning Tree. See [“How MSTP Works” on page 10-4](#) for more information.
- **Configure an MST region name and revision level.** Switches that share the same MST region name, revision level, and VLAN to Multiple Spanning Tree Instance (MSTI) mapping belong to the same MST region. See [“What is a Multiple Spanning Tree Region” on page 10-8](#) for more information.
- **Configure MSTIs.** By default, every switch has a Common and Internal Spanning Tree (CIST) instance 0, which is also referred to as MSTI 0. Configuration of additional MSTI is required to segment switch VLANs into separate instances. See [“What is a Multiple Spanning Tree Instance \(MSTI\)” on page 10-7](#) for more information.
- **Map VLANs to MSTI.** By default, all existing VLANs are mapped to the CIST instance 0. Associating a VLAN to an MSTI specifies which Spanning Tree instance will determine the best data path for traffic carried on the VLAN. In addition, the VLAN-to-MSTI mapping is also one of three MST configuration attributes used to determine that the switch belongs to a particular MST region.

For a tutorial on setting up an example MST configuration, see [“Quick Steps for Configuring an MST Region” on page 10-14](#) and [“Quick Steps for Configuring MSTIs” on page 10-16](#).

Using Spanning Tree Configuration Commands

The Alcatel-Lucent implementation of the Multiple Spanning Tree Protocol introduces the concept of *implicit* and *explicit* CLI commands for Spanning Tree configuration and verification. Explicit commands contain one of the following keywords that specifies the type of Spanning Tree instance to modify:

- **cist**—command applies to the Common and Internal Spanning Tree instance.
- **msti**—command applies to the specified Multiple Spanning Tree Instance.
- **1x1**—command applies to the specified VLAN instance.

Explicit commands allow the configuration of a particular Spanning Tree instance independent of which mode and/or protocol is currently active on the switch. The configuration, however, does not go active until the switch is changed to the appropriate mode. For example, if the switch is running in the 1x1 mode, the following explicit commands changes the MSTI 3 priority to 12288:

```
-> bridge msti 3 priority 12288
```

Even though the above command is accepted in the 1x1 mode, the new priority value does not take effect until the switch mode is changed to flat mode.

Note that explicit commands using the **cist** and **msti** keywords are required to define an MSTP configuration. Implicit commands are only allowed for defining STP or RSTP configurations.

Implicit commands resemble previously implemented Spanning Tree commands, but apply to the appropriate instance based on the current mode and protocol that is active on the switch. For example, if the 1x1 mode is active, the instance number specified with the following command implies a VLAN ID:

```
-> bridge 255 priority 16384
```

If the flat mode is active, the single flat mode instance is implied and thus configured by the command. Since the flat mode instance is implied in this case, there is no need to specify an instance number. For example, the following command configures the protocol for the flat mode instance:

```
-> bridge protocol mstp
```

Similar to previous releases, it is possible to configure the flat mode instance by specifying **1** for the instance number (e.g., **bridge 1 protocol rstp**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

Note. When a snapshot is taken of the switch configuration, the explicit form of all Spanning Tree commands is captured. For example, if the priority of MSTI 2 was changed from the default value to a priority of 16384, then **bridge msti 2 priority 16384** is the command captured to reflect this in the snapshot file. In addition, explicit commands are captured for both flat and 1x1 mode configurations.

For more information about Spanning Tree configuration commands as they apply to all supported protocols (STP, RSTP, and MSTP), see [Chapter 11, “Configuring Spanning Tree Parameters.”](#)

Understanding Spanning Tree Modes

The switch can operate in one of two Spanning Tree modes: *flat* and *1x1*. The flat mode provides a Common Spanning Tree (CST) instance that applies across all VLANs by default. This mode supports the use of the STP (802.1D), RSTP (802.1w), and MSTP. MSTP allows the mapping of one or more VLANs to a single Spanning Tree instance.

The 1x1 mode is an Alcatel-Lucent proprietary implementation that automatically calculates a separate Spanning Tree instance for each VLAN configured on the switch. This mode only supports the use of the STP and RSTP protocols.

Although MSTP is not supported in the 1x1 mode, it is possible to define an MSTP configuration in this mode using explicit Spanning Tree commands. See [“Using Spanning Tree Configuration Commands” on page 10-10](#) for more information about explicit commands.

By default, a switch is running in the 1x1 mode and using the 802.1D protocol when it is first turned on. See [Chapter 11, “Configuring Spanning Tree Parameters,”](#) for more information about Spanning Tree modes.

MST Interoperability and Migration

Connecting an MSTP switch to a non-MSTP flat mode switch is supported. Since the Common and Internal Spanning Tree (CIST) controls the flat mode instance on both switches, STP or RSTP can remain active on the non-MSTP switch within the network topology.

An MSTP switch is part of a Multiple Spanning Tree (MST) Region, which appears as a single, flat mode instance to the non-MSTP switch. The port that connects the MSTP switch to the non-MSTP switch is referred to as a *boundary* port. When a boundary port detects an STP (802.1D) or RSTP (802.1w) BPDU, it responds with the appropriate protocol BPDU to provide interoperability between the two switches. This interoperability also serves to indicate the edge of the MST region.

Interoperability between MSTP switches and 1x1 mode switches is not recommended. The 1x1 mode is a proprietary implementation that creates a separate Spanning Tree instance for each VLAN configured on the switch. The MSTP implementation is in compliance with the IEEE standard and is only supported on flat mode switches.

Tagged BPDU transmitted from a 1x1 switch are ignored by a flat mode switch, which can cause a network loop to go undetected. Although it is not recommended, it may be necessary to temporarily connect a 1x1 switch to a flat mode switch until migration to MSTP is complete. If this is the case, then only configure a fixed, untagged connection between VLAN 1 on both switches.

Migrating from Flat Mode STP/RSTP to Flat Mode MSTP

Migrating an STP/RSTP flat mode switch to MSTP is relatively transparent. When STP or RSTP is the active protocol, the Common and Internal Spanning Tree (CIST) controls the flat mode instance. If on the same switch the protocol is changed to MSTP, the CIST still controls the flat mode instance.

Note the following when converting a flat mode STP/RSTP switch to MSTP:

- Making a backup copy of the switch **boot.cfg** file before changing the protocol to MSTP is highly recommended. Having a backup copy will make it easier to revert to the non-MSTP configuration if necessary. Once MSTP is active, commands are written in their explicit form and not compatible with previous releases of Spanning Tree.
- When converting multiple switches, change the protocol to MSTP first on every switch before starting to configure Multiple Spanning Tree Instances (MSTI).
- Once the protocol is changed, MSTP features are available for configuration. Multiple Spanning Tree Instances (MSTI) are now configurable for defining data paths for VLAN traffic. See [“How MSTP Works” on page 10-4](#) for more information.
- Using explicit Spanning Tree commands to define the MSTP configuration is required. Implicit commands are for configuring STP and RSTP. See [“Using Spanning Tree Configuration Commands” on page 10-10](#) for more information.
- STP and RSTP use a 16-bit port path cost (PPC) and MSTP uses a 32-bit PPC. When the protocol is changed to MSTP, the bridge priority and PPC values for the flat mode CIST instance are reset to their default values.
- It is possible to configure the switch to use 32-bit PPC value for all protocols (see the [bridge path cost mode](#) command page for more information). If this is the case, then the PPC for the CIST is not reset when the protocol is changed to/from MSTP.
- This implementation of MSTP is compliant with the IEEE 802.1Q 2005 standard and thus provides interconnectivity with MSTP compliant systems.

Migrating from 1x1 Mode to Flat Mode MSTP

As previously described, the 1x1 mode is an Alcatel-Lucent proprietary implementation that applies one Spanning Tree instance to each VLAN. For example, if five VLANs exist on the switch, then there are five Spanning Tree instances active on the switch, unless Spanning Tree is disabled on one of the VLANs.

Note the following when converting a 1x1 mode STP/RSTP switch to flat mode MSTP:

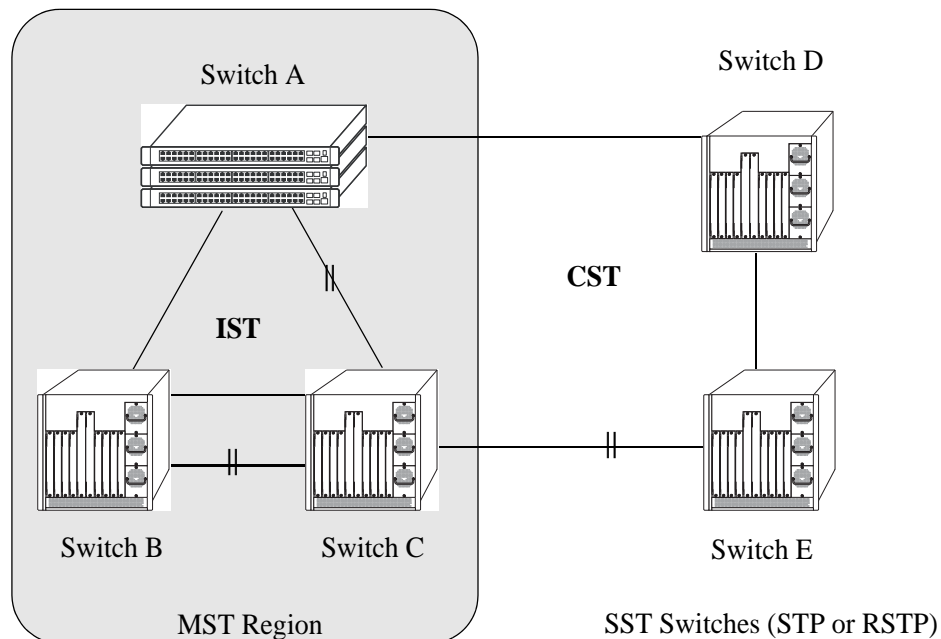
- Making a backup copy of the switch **boot.cfg** file before changing the protocol to MSTP is highly recommended. Having a backup copy will make it easier to revert to the non-MSTP configuration if necessary. Once MSTP is active, commands are written in their explicit form and not compatible with previous releases of Spanning Tree.
- Using MSTP requires changing the switch mode from 1x1 to flat. When the mode is changed from 1x1 to flat, ports still retain their VLAN associations but are now part of a single, flat mode Spanning Tree instance that spans across all VLANs. As a result, a path that was forwarding traffic in the 1x1 mode may transition to a blocking state after the mode is changed to flat.
- Once the protocol is changed, MSTP features are available for configuration. Multiple Spanning Tree Instances (MSTI) are now configurable for defining data paths for VLAN traffic. See [“How MSTP Works” on page 10-4](#) for more information.
- Note that STP/RSTP use a 16-bit port path cost (PPC) and MSTP uses a 32-bit PPC. When the protocol is changed to MSTP, the bridge priority and PPC values for the flat mode CIST instance are reset to their default values.
- It is possible to configure the switch to use 32-bit PPC value for all protocols (see the [bridge path cost mode](#) command page for more information). If this is the case, then the PPC for the CIST is not reset when the protocol is changed to/from MSTP.
- This implementation of MSTP is compliant with the IEEE 802.1Q 2005 standard and thus provides interconnectivity with MSTP compliant systems.

Quick Steps for Configuring an MST Region

An MST region identifies a group of MSTP switches that is seen as a single, flat mode instance by other regions and/or non-MSTP switches. A region is defined by three attributes: name, revision level, and a VLAN-to-MSTI mapping. Switches configured with the same value for all three of these attributes belong to the same MST region.

Note that an additional configurable MST region parameter defines the maximum number of hops authorized for the region but is not considered when determining regional membership. The maximum hops value is the value used by all bridges within the region when the bridge is acting as the root of the MST region.

This section provides a tutorial for defining a sample MST region configuration, as shown in the diagram below:



In order for switches A, B, and C in the above diagram to belong to the same MST region, they must all share the same values for region name, revision level, and configuration digest (VLAN-to-MSTI mapping).

The following steps are performed on each switch to define **Alcatel-Lucent Marketing** as the MST region name, **2000** as the MST region revision level, map existing VLANs to existing MSTIs, and **3** as the maximum hops value for the region:

- 1 Configure an MST Region name using the **bridge mst region name** command. For example:

```
-> bridge mst region name "Alcatel Marketing"
```

- 2 Configure the MST Region revision level using the **bridge mst region revision level** command. For example:

```
-> bridge mst region revision level 2000
```

3 Map VLANs 100 and 200 to MSTI 2 and VLANs 300 and 400 to MSTI 4 using the **bridge msti vlan** command to define the configuration digest. For example:

```
-> bridge msti 2 vlan 100 200
-> bridge msti 4 vlan 300 400
```

See “[Quick Steps for Configuring MSTIs](#)” on page 10-16 for a tutorial on how to create and map MSTIs to VLANs.

4 Configure **3** as the maximum number of hops for the region using the **bridge mst region max hops** command. For example:

```
-> bridge mst region max hops 3
```

Note. (*Optional*) Verify the MST region configuration on each switch with the **show spantree mst region** command. For example:

```
-> show spantree mst region
Configuration Name      : Alcatel Marketing,
Revision Level         : 2000,
Configuration Digest   : 0x922fb3f 31752d68 67fe1155 d0ce8380,
Revision Max hops     : 3,
Cist Instance Number   : 0
```

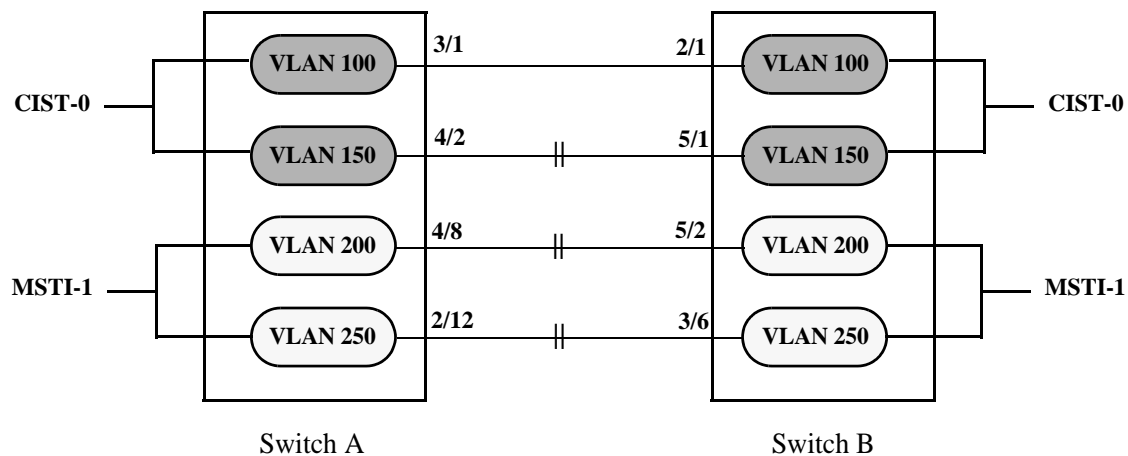
All switches configured with the exact same values as shown in the above example are considered members of the Alcatel-Lucent Marketing MST region.

Quick Steps for Configuring MSTIs

By default, the Spanning Tree software is active on all switches and operating in the 1x1 mode using 802.1w RSTP. A loop-free network topology is automatically calculated based on default 802.1w RSTP switch, bridge, and port parameter values.

Using Multiple Spanning Tree (MST) requires configuration changes to the default Spanning Tree values (mode and protocol) as well as defining specific MSTP parameters and instances.

The following steps provide a tutorial for setting up a sample MSTP configuration, as shown in the diagram below:



Flat Mode MSTP Quick Steps Example

1 Change the Spanning Tree operating mode, if necessary, on Switch A and Switch B from 1x1 to flat mode using the **bridge mode** command. For example:

```
-> bridge mode flat
```

Note that defining an MSTP configuration requires the use of explicit Spanning Tree commands, which are available in both the flat and 1x1 mode. As a result, this step is optional. See [“Using Spanning Tree Configuration Commands” on page 10-10](#) for more information.

2 Change the Spanning Tree protocol to MSTP using the **bridge protocol** command. For example:

```
-> bridge protocol mstp
```

3 Create VLANs 100, 200, 300, and 400 using the **vlan** command. For example:

```
-> vlan 100
-> vlan 150
-> vlan 200
-> vlan 250
```

4 Assign switch ports to VLANs, as shown in the above diagram, using the **vlan port default** command. For example, the following commands assign ports 3/1, 4/2, 4/8, and 2/12 to VLANs 100, 150, 200, and 250 on Switch A:

```
-> vlan 100 port default 3/1
-> vlan 150 port default 4/2
-> vlan 200 port default 4/8
-> vlan 250 port default 2/12
```

The following commands assign ports 2/1, 5/1, 5/2, and 3/6 to VLANs 100, 150, 200, and 250 on Switch B:

```
-> vlan 100 port default 2/1
-> vlan 150 port default 5/1
-> vlan 200 port default 5/2
-> vlan 250 port default 3/6
```

5 Create one MSTI using the **bridge msti** command. For example:

```
-> bridge msti 1
```

6 Assign VLANs 200 and 250 to MSTI 1. For example:

```
-> bridge msti 1 vlan 100 200
```

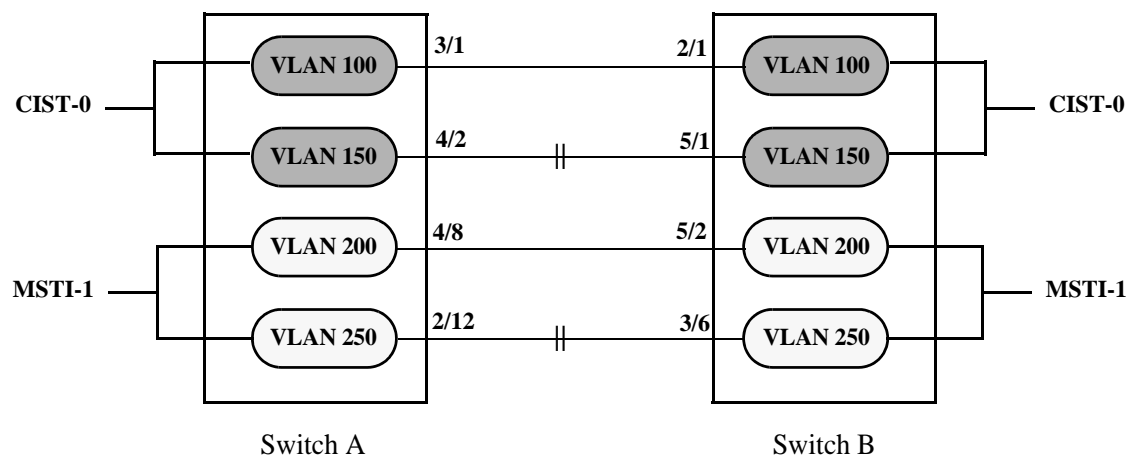
By default, all VLANs are associated with the CIST instance. As a result, VLANs 100 and 150 do not require any configuration to map them to the CIST instance.

7 Configure the port path cost (PPC) for all ports on both switches associated with MSTI 1 to a PPC value that is lower than the PPC value for the ports associated with the CIST instance using the **bridge msti slot/port path cost** command. For example, the PPC for ports associated with the CIST instance is set to the default of 200,000 for 100 MB connections. The following commands change the PPC value for ports associated with the MSTI 1 to 20,000:

```
-> bridge msti 1 4/8 path cost 20,000
-> bridge msti 1 2/12 path cost 20,000
-> bridge msti 1 5/2 path cost 20,000
-> bridge msti 1 3/6 path cost 20,000
```

Note that in this example, port connections between VLANs 150, 200, and 250 on each switch initially were blocked, as shown in the diagram on [page 10-16](#). This is because in flat mode MSTP, each instance is active on all ports resulting in a comparison of connections independent of VLAN and MSTI associations.

To avoid this and allow VLAN traffic to flow over separate data paths based on MSTI association, Step 7 of this tutorial configures a superior port path cost value for ports associated with MSTI 1. As a result, MSTI 1 selects one of the data paths between its VLANs as the best path, rather than the CIST data paths, as shown in the diagram on [page 10-18](#).



Flat Mode MSTP with Superior MSTI 1 PPC Values

Note that of the two data paths available to MSTI 1 VLANs, one is still blocked because it is seen as redundant for that instance. In addition, the CIST data path still remains available for CIST VLAN traffic.

Another solution to this scenario is to assign all VLANs to an MSTI, leaving no VLANs controlled by the CIST. As a result, the CIST BPDU will only contain MSTI information. See [“How MSTP Works” on page 10-4](#) for more information.

Verifying the MST Configuration

To display information about the MST configuration on the switch, use the show commands listed below:

show spantree cist	Displays the Spanning Tree bridge configuration for the flat mode Common and Internal Spanning Tree (CIST) instance.
show spantree msti	Displays Spanning Tree bridge information for a Multiple Spanning Tree Instance (MSTI).
show spantree cist ports	Displays Spanning Tree port information for the flat mode Common and Internal Spanning Tree (CIST) instance.
show spantree msti ports	Displays Spanning Tree port information for a flat mode Multiple Spanning Tree Instance (MSTI).
show spantree mst region	Displays the Multiple Spanning Tree (MST) region information for the switch.
show spantree cist vlan-map	Displays the range of VLANs associated with the flat mode Common and Internal Spanning Tree (CIST) instance.
show spantree msti vlan-map	Displays the range of VLANs associated with the specified Multiple Spanning Tree Instance (MSTI).
show spantree map-msti	Displays the Multiple Spanning Tree Instance (MSTI) that is associated to the specified VLAN.
show spantree mst port	Displays a summary of Spanning Tree connection information and instance associations for the specified port or a link aggregate of ports.

For more information about the resulting displays from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

11 Configuring Spanning Tree Parameters

The Spanning Tree Algorithm and Protocol (STP) is a self-configuring algorithm that maintains a loop-free topology while providing data path redundancy and network scalability. Based on the IEEE 802.1D standard, the Alcatel-Lucent STP implementation distributes the Spanning Tree load between the primary management module and the network interface modules. In the case of a stack of switches, the STP load is distributed between the primary management switch and other switches in the stack. This functionality improves network robustness by providing a Spanning Tree that continues to respond to BPDUs (Bridge Protocol Data Unit) and port link up and down states in the event of a fail over to a backup management module or switch.

The Alcatel-Lucent distributed implementation also incorporates the following Spanning Tree features:

- Configures a physical topology into a single Spanning Tree to ensure that there is only one data path between any two switches.
- Supports fault tolerance within the network topology. The Spanning Tree is configured again in the event of a data path or bridge failure or when a new switch is added to the topology.
- Supports two Spanning Tree operating modes; *flat* (single STP instance per switch) and *1x1* (single STP instance per VLAN). The 1x1 mode can be configured to interoperate with Cisco's proprietary Per VLAN Spanning Tree instance (PVST+).
- Supports four Spanning Tree Algorithms; 802.1D (STP), 802.1w (RSTP), 802.1Q 2005 (MSTP), and RRSTP.
- Allows 802.1Q tagged ports and link aggregate logical ports to participate in the calculation of the STP topology.

The Distributed Spanning Tree software is active on all switches by default. As a result, a loop-free network topology is automatically calculated based on default Spanning Tree switch, VLAN, and port parameter values. It is only necessary to configure Spanning Tree parameters to change how the topology is calculated and maintained.

In This Chapter

This chapter provides an overview about how Spanning Tree works and how to configure Spanning Tree parameters through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Selecting the switch Spanning Tree operating mode (flat or 1x1) on [page 11-12](#).
- Configuring Spanning Tree bridge parameters on [page 11-17](#).
- Configuring Spanning Tree port parameters on [page 11-26](#).
- Configuring Ring Rapid Spanning Tree on [page 11-39](#).
- Configuring an example Spanning Tree topology on [page 11-40](#).

Spanning Tree Specifications

IEEE Standards supported	802.1D– <i>Media Access Control (MAC) Bridges</i> 802.1w– <i>Rapid Reconfiguration (802.1D Amendment 2)</i> 802.1Q 2005– <i>Virtual Bridged Local Area Networks</i> 802.1Q 2005– <i>Multiple Spanning Trees (MSTP)</i>
Spanning Tree Protocols supported	802.1D Standard Spanning Tree Algorithm and Protocol (STP) 802.1w Rapid Spanning Tree Algorithm and Protocol (RSTP) 802.1Q 2005 Multiple Spanning Tree Protocol (MSTP) Ring Rapid Spanning Tree Protocol (RRSTP)
Platforms Supported	OmniSwitch 6250 Series
Spanning Tree Operating Modes supported	Flat mode - one spanning tree instance per switch 1x1 mode - one spanning tree instance per VLAN
Spanning Tree port eligibility	Fixed ports (non-mobile) 802.1Q tagged ports Link aggregate of ports
Number of 1x1 Spanning Tree instances supported	252
Number of Multiple Spanning Tree Instances (MSTI) supported	16 MSTI, in addition to the Common and Internal Spanning Tree instance (also referred to as MSTI 0).
Number of Ring Rapid Spanning Tree (RRSTP) rings supported	8
CLI Command Prefix Recognition	All Spanning Tree commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

Spanning Tree Bridge Parameter Defaults

Parameter Description	Command	Default
Spanning Tree operating mode	bridge mode	1x1 (a separate Spanning Tree instance for each VLAN)
PVST+ status	bridge mode 1x1 pvst+	Disabled
Spanning Tree protocol	bridge protocol	RSTP (802.1w)
BPDU switching status	bridge bpdu-switching	Disabled
Priority value for the Spanning Tree instance	bridge priority	32768
Hello time interval between each BPDU transmission	bridge hello time	2 seconds
Maximum aging time allowed for Spanning Tree information learned from the network	bridge max age	20 seconds
Spanning Tree port state transition time	bridge forward delay	15 seconds
Automatic VLAN Containment	bridge auto-vlan-containment	Disabled

Spanning Tree Port Parameter Defaults

Parameter Description	Command	Default
Spanning Tree port administrative state	bridge slot/port	Enabled
Spanning Tree port priority value	bridge slot/port priority	7
Spanning Tree port path cost	bridge slot/port path cost	0 (cost is based on port speed)
Path cost mode	bridge path cost mode	Auto (16-bit in 1x1 mode and STP or RSTP flat mode, 32-bit in MSTP flat mode)
Port state management mode	bridge slot/port mode	Dynamic (Spanning Tree Algorithm determines port state)
Type of port connection	bridge slot/port connection	auto point to point
Type of BPDU to be used on a port when 1X1 PVST+ mode is enabled	bridge port pvst+	auto (IEEE BPDUs are used until a PVST+ BPDU is detected)

Multiple Spanning Tree (MST) Region Defaults

Although the following parameter values are specific to MSTP, they are configurable regardless of which mode (flat or 1x1) or protocol is active on the switch.

Parameter Description	Command	Default
The MST region name	bridge mst region name	blank
The revision level for the MST region	bridge mst region revision level	0
The maximum number of hops authorized for the region	bridge mst region max hops	20
The number of Multiple Spanning Tree Instances (MSTI)	bridge msti	1 (flat mode instance)
The VLAN to MSTI mapping	bridge msti vlan	All VLANs are mapped to the Common Internal Spanning Tree (CIST) instance

Ring Rapid Spanning Tree Defaults

The following parameter value is specific to RRSTP and is only configurable when the flat mode is active on the switch.

Parameter Description	Command	Default
Ring Rapid Spanning Tree Protocol status	bridge rrstp	Disabled
Number of rings	bridge rrstp ring	0
Ring status	bridge rrstp ring bridge rrstp ring status	Disabled

Spanning Tree Overview

Alcatel-Lucent switches support the use of the 802.1D Spanning Tree Algorithm and Protocol (STP), the 802.1w Rapid Spanning Tree Algorithm and Protocol (RSTP), the 802.1Q 2005 Multiple Spanning Tree Protocol (MSTP), and the Ring Rapid Spanning Tree Protocol (RRSTP).

RSTP expedites topology changes by allowing blocked ports to transition directly into a forwarding state, bypassing listening and learning states. This provides rapid reconfiguration of the Spanning Tree in the event of a network path or device failure.

The 802.1w standard is an amendment to the 802.1D document, thus RSTP is based on STP. Regardless of which one of these two protocols a switch or VLAN is running, it can successfully interoperate with other switches or VLANs.

802.1Q 2005 is a new version of MSTP that combines the 802.1D 2004 and 802.1S protocols. This implementation of 802.1Q 2005 also includes improvements to edge port configuration and provides administrative control to restrict port role assignment and the propagation of topology change information through bridge ports.

MSTP is an enhancement to the 802.1Q Common Spanning Tree (CST), which is provided when an Alcatel-Lucent switch is running in the flat Spanning Tree operating mode. The flat mode applies a single spanning tree instance across all VLAN port connections on a switch. MSTP allows the configuration of Multiple Spanning Tree Instances (MSTIs) in addition to the CST instance. Each MSTI is mapped to a set of VLANs. As a result, flat mode can now support the forwarding of VLAN traffic over separate data paths.

RRSTP is faster than MSTP. It is used in a ring topology where bridges are connected in a point to point manner. This protocol identifies the bridge hosting the alternate (ALT) port in lesser convergence time. This ALT port is changed to the forwarding state immediately without altering the MSTP state to enable the data path. The RRSTP frame travels from the point of failure to the bridge hosting the ALT port in both the directions. The MAC addresses matching the ports in the ring are flushed to make the data path convergence much faster than normal MSTP.

This section provides a Spanning Tree overview based on RSTP operation and terminology. Although MSTP is based on RSTP, see [Chapter 10, “Using 802.1Q 2005 Multiple Spanning Tree,”](#) for specific information about configuring MSTP. For more information about using RRSTP, see [“Using RRSTP” on page 11-38.](#)

How the Spanning Tree Topology is Calculated

The *tree* consists of links and bridges that provide a single data path that spans the bridged network. At the base of the tree is a *root bridge*. One bridge is elected by all the bridges participating in the network to serve as the root of the tree. After the root bridge is identified, STP calculates the best path that leads from each bridge back to the root and blocks any connections that would cause a network loop.

To determine the best path to the root, STP uses the *path cost* value, which is associated with every port on each bridge in the network. This value is a configurable weighted measure that indicates the contribution of the port connection to the entire path leading from the bridge to the root.

In addition, a *root path cost* value is associated with every bridge. This value is the sum of the path costs for the port that receives frames on the best path to the root (this value is zero for the root bridge). The bridge with the lowest root path cost becomes the *designated bridge* for the LAN, as it provides the shortest path to the root for all bridges connected to the LAN.

During the process of calculating the Spanning Tree topology, each port on every bridge is assigned a *port role* based on how the port and/or its bridge will participate in the active Spanning Tree topology.

The following table provides a list of port role types and the port and/or bridge properties that the Spanning Tree Algorithm examines to determine which role to assign to the port.

Role	Port/Bridge Properties
Root Port	Port connection that provides the shortest path (lowest path cost value) to the root. The root bridge does not have a root port.
Designated Port	The designated bridge provides the LAN with the shortest path to the root. The designated port connects the LAN to this bridge.
Backup Port	Any operational port on the designated bridge that is not a root or designated port. Provides a backup connection for the designated port. A backup port can only exist when there are redundant designated port connections to the LAN.
Alternate Port	Any operational port that is not the root port for its bridge and its bridge is not the designated bridge for the LAN. An alternate port offers an alternate path to the root bridge if the root port on its own bridge goes down.
Disabled Port	Port is not operational. If an active connection does come up on the port, it is assigned an appropriate role.

Note. The distinction between a backup port and an alternate port was introduced with the IEEE 802.1w standard to help define rapid transition of an alternate port to a root port.

The role a port plays or may potentially play in the active Spanning Tree topology determines the port's operating state; *discarding*, *learning*, or *forwarding*. The *port state* is also configurable in that it is possible to enable or disable a port's administrative status and/or specify a forwarding or blocking state that is only changed through user intervention.

The Spanning Tree Algorithm only includes ports in its calculations that are operational (link is up) and have an enabled administrative status. The following table compares and defines 802.1D and 802.1w port states and their associated port roles:

STP Port State	RSTP Port State	Port State Definition	Port Role
Disabled	Discarding	Port is down or administratively disabled and is not included in the topology.	Disabled
Blocking	Discarding	Frames are dropped, nothing is learned or forwarded on the port. Port is temporarily excluded from topology.	Alternate, Backup
Learning	Learning	Port is learning MAC addresses that are seen on the port and adding them to the bridge forwarding table, but not transmitting any data. Port is included in the active topology.	Root, Designated
Forwarding	Forwarding	Port is transmitting and receiving data and is included in the active topology.	Root, Designated

Once the Spanning Tree is calculated, there is only one root bridge, one designated bridge for each LAN, and one root port on each bridge (except for the root bridge). Data travels back and forth between bridges over forwarding port connections that form the best, non-redundant path to the root. The active topology ensures that network loops do not exist.

Bridge Protocol Data Units (BPDU)

Switches send layer 2 frames, referred to as Configuration Bridge Protocol Data Units (BPDU), to relay information to other switches. The information in these BPDU is used to calculate and reconfigure the Spanning Tree topology. A Configuration BPDU contains the following information that pertains to the bridge transmitting the BPDU:

Root ID	The Bridge ID for the bridge that this bridge believes is the root.
Root Path Cost	The sum of the Path Costs that lead from the root bridge to this bridge port. The Path Cost is a configurable parameter value. The IEEE 802.1D standard specifies a default value that is based on port speed. See “Configuring Port Path Cost” on page 11-31 for more information.
Bridge ID	An eight-byte hex value that identifies this bridge within the Spanning Tree. The first two bytes contain a configurable priority value and the remaining six bytes contain a bridge MAC address. See “Configuring the Bridge Priority” on page 11-20 for more information. Each switch chassis is assigned a dedicated base MAC address. This is the MAC address that is combined with the priority value to provide a unique Bridge ID for the switch. For more information about the base MAC address, see the appropriate Hardware Users Guide for the switch.
Port ID	A 16-bit hex value that identifies the bridge port that transmitted this BPDU. The first 4 bits contain a configurable priority value and the remaining 12 bits contain the physical switch port number. See “Configuring Port Priority” on page 11-30 for more information.

The sending and receiving of Configuration BPDU between switches participating in the bridged network constitute the root bridge election; the best path to the root is determined and then advertised to the rest of the network. BPDU provide enough information for the STP software running on each switch to determine the following:

- Which bridge will serve as the root bridge.
- The shortest path between each bridge and the root bridge.
- Which bridge will serve as the designated bridge for the LAN.
- Which port on each bridge will serve as the root port.
- The port state (forwarding or discarding) for each bridge port based on the role the port will play in the active Spanning Tree topology.

The following events trigger the transmitting and/or processing of BPDU in order to discover and maintain the Spanning Tree topology:

- When a bridge first comes up, it assumes it is the root and starts transmitting Configuration BPDU on all its active ports advertising its own bridge ID as the root bridge ID.

- When a bridge receives BPDU on its root port that contains more attractive information (higher priority parameters and/or lower path costs), it forwards this information on to other LANs to which it is connected for consideration.
- When a bridge receives BPDU on its designated port that contains information that is less attractive (lower priority values and/or higher path costs), it forwards its own information to other LANs to which it is connected for consideration.

STP evaluates BPDU parameter values to select the best BPDU based on the following order of precedence:

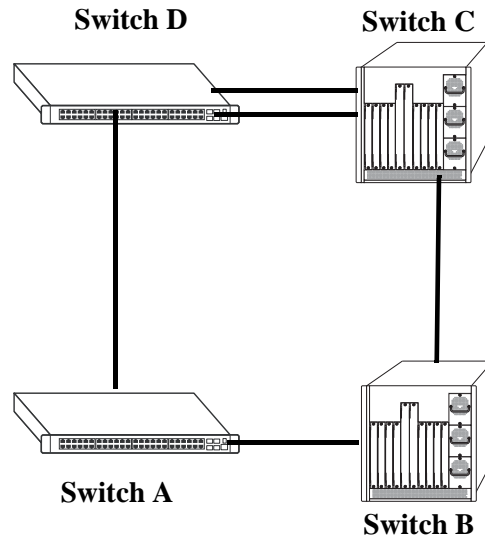
- 1** The lowest root bridge ID (lowest priority value, then lowest MAC address).
- 2** The best root path cost.
- 3** If root path costs are equal, the bridge ID of the bridge sending the BPDU.
- 4** If the previous three values tie, then the port ID (lowest priority value, then lowest port number).

When a topology change occurs, such as when a link goes down or a switch is added to the network, the affected bridge sends Topology Change Notification (TCN) BPDU to the designated bridge for its LAN. The designated bridge will then forward the TCN to the root bridge. The root then sends out a Configuration BPDU and sets a Topology Change (TC) flag within the BPDU to notify other bridges that there is a change in the configuration information. Once this change is propagated throughout the Spanning Tree network, the root stops sending BPDU with the TC flag set and the Spanning Tree returns to an active, stable topology.

Note. You can restrict the propagation of TCNs on a port. To restrict TCN propagation on a port, see [“Configuring STP Port Parameters”](#) on page 11-26.

Topology Examples

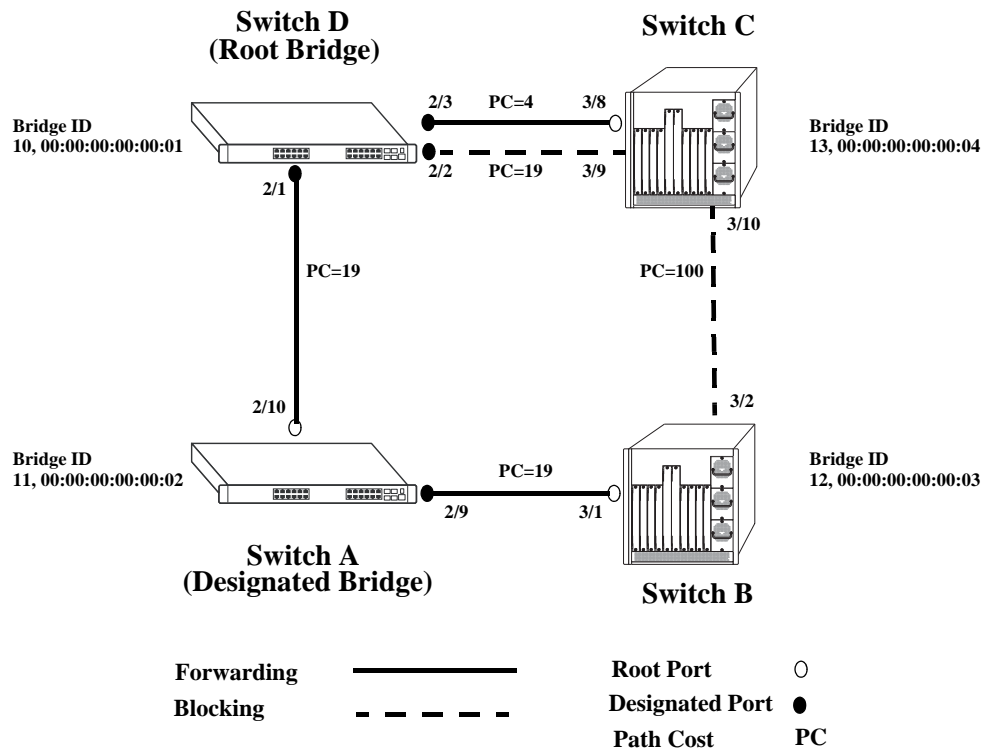
The following diagram shows an example of a physical network topology that incorporates data path redundancy to ensure fault tolerance. These redundant paths, however, create loops in the network configuration. If a device connected to Switch A sends broadcast packets, Switch A will flood the packets out all of its active ports. The switches connected to Switch A will in turn flood the broadcast packets out their active ports, and Switch A will eventually receive the same packets back and the cycle will start over again. This causes severe congestion on the network, often referred to as a *broadcast storm*.



Physical Topology Example

The Spanning Tree Algorithm prevents network loops by ensuring that there is always only one active link between any two switches. This is done by transitioning one of the redundant links into a blocking state, leaving only one link actively forwarding traffic. If the active link goes down, then Spanning Tree will transition one of the blocked links to the forwarding state to take over for the downed link. If a new switch is added to the network, the Spanning Tree topology is automatically recalculated to include the monitoring of links to the new switch.

The following diagram shows the logical connectivity of the same physical topology as determined by the Spanning Tree Algorithm:



Active Spanning Tree Topology Example

In the above active Spanning Tree topology example, the following configuration decisions were made as a result of calculations performed by the Spanning Tree Algorithm:

- Switch D is the root bridge because its bridge ID has a priority value of 10 (the lower the priority value, the higher the priority the bridge has in the Spanning Tree). If all four switches had the same priority, then the switch with the lowest MAC address in its bridge ID would become the root.
- Switch A is the designated bridge for Switch B, because it provides the best path for Switch B to the root bridge.
- Port 2/9 on Switch A is a designated port, because it connects the LAN from Switch B to Switch A.
- All ports on Switch D are designated ports, because Switch D is the root and each port connects to a LAN.
- Ports 2/10, 3/1, and 3/8 are the root ports for Switches A, B, and C, respectively, because they offer the shortest path towards the root bridge.
- The port 3/9 connection on Switch C to port 2/2 on Switch D is in a discarding (blocking) state, as the connection these ports provides is redundant (backup) and has a higher path cost value than the 2/3 to 3/8 connection between the same two switches. As a result, a network loop is avoided.
- The port 3/2 connection on Switch B to port 3/10 on Switch C is also in a discarding (blocking) state, as the connection these ports provides has a higher path cost to root Switch D than the path between Switch B and Switch A. As a result, a network loop is avoided.

Spanning Tree Operating Modes

The switch can operate in one of two Spanning Tree modes: *flat* and *1x1*. Both modes apply to the entire switch and determine whether a single Spanning Tree instance is applied across multiple VLANs (flat mode) or a single instance is applied to each VLAN (1x1 mode). By default, a switch is running in the 1x1 mode when it is first turned on.

Use the **bridge mode** command to select the flat or 1x1 Spanning Tree mode. The switch operates in one mode or the other, however, it is not necessary to reboot the switch when changing modes. To determine which mode the switch is operating in, use the **bridge rstp ring vlan-tag** command. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Using Flat Spanning Tree Mode

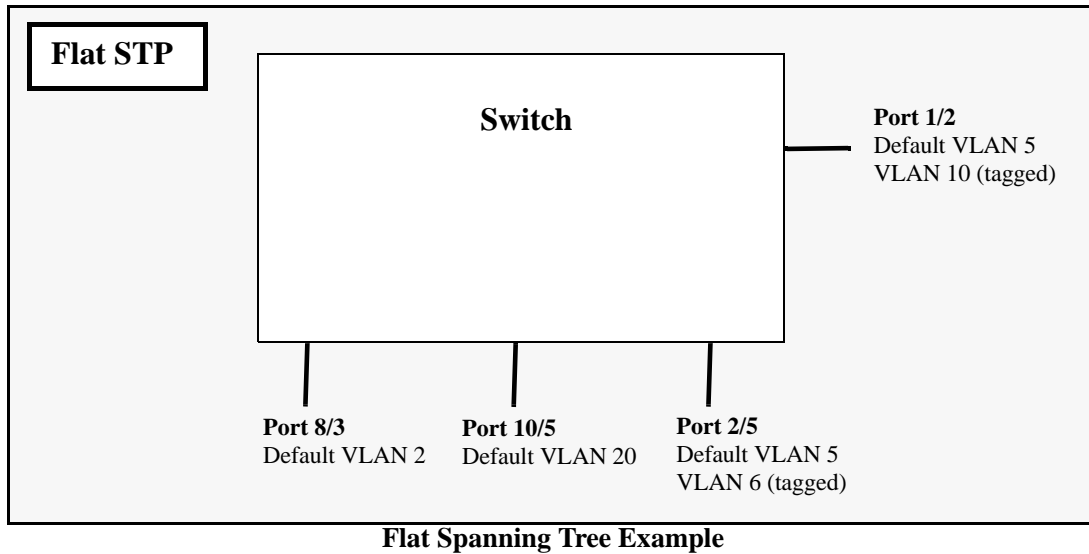
Before selecting the flat Spanning Tree mode, consider the following:

- If STP (802.1D) is the active protocol, then there is one Spanning Tree instance for the entire switch; port states are determined across VLANs. If MSTP (802.1s) is the active protocol, then multiple instances up to a total of 17 are allowed. Port states, however, are still determined across VLANs.
- Multiple connections between switches are considered redundant paths even if they are associated with different VLANs.
- Spanning Tree parameters are configured for the single flat mode instance. For example, if Spanning Tree is disabled on VLAN 1, then it is disabled for all VLANs. Disabling STP on any other VLAN, however, only exclude ports associated with that VLAN from the Spanning Tree Algorithm.
- Fixed (untagged) and 802.1Q tagged ports are supported in each VLAN. BPDU, however, are always untagged.
- When the Spanning Tree mode is changed from 1x1 to flat, ports still retain their VLAN associations but are now part of a single Spanning Tree instance that spans across all VLANs. As a result, a path that was forwarding traffic in the 1x1 mode may transition to a blocking state after the mode is changed to flat.

To change the Spanning Tree operating mode to flat, enter the following command:

```
-> bridge mode flat
```

The following diagram shows a flat mode switch with STP (802.1D) as the active protocol. All ports, regardless of their default VLAN configuration or tagged VLAN assignments, are considered part of one Spanning Tree instance. To see an example of a flat mode switch with MSTP (802.1s) as the active protocol, see [Chapter 10, “Using 802.1Q 2005 Multiple Spanning Tree.”](#)



In the above example, if port 8/3 connects to another switch and port 10/5 connects to that same switch, the Spanning Tree Algorithm would detect a redundant path and transition one of the ports into a blocking state. The same holds true for the tagged ports.

Using 1x1 Spanning Tree Mode

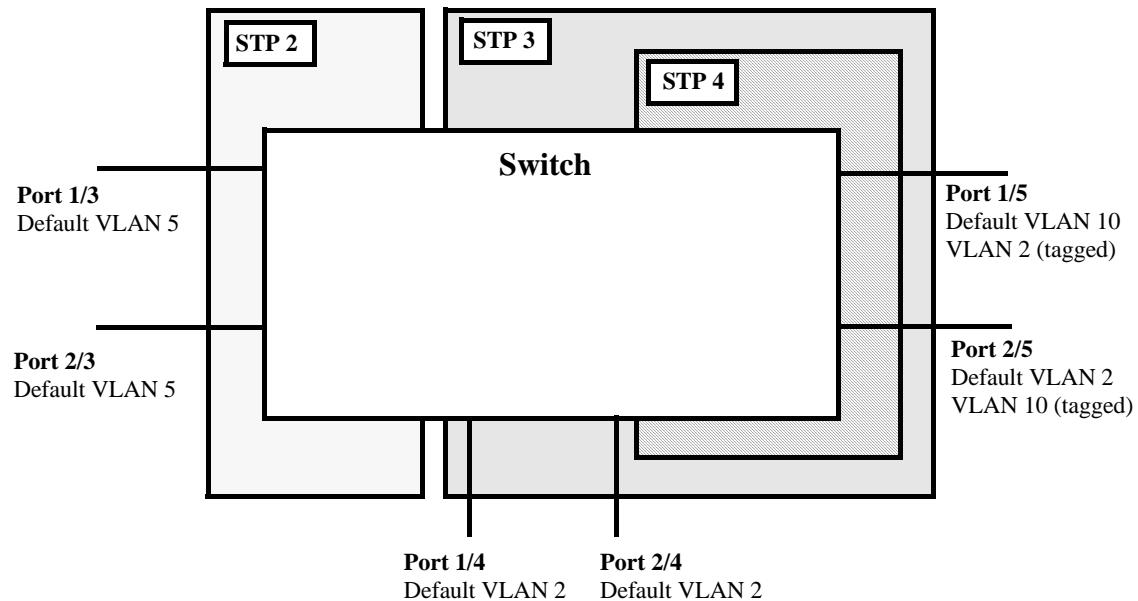
Before selecting the 1x1 Spanning Tree operating mode, consider the following:

- A single Spanning Tree instance is enabled for each VLAN configured on the switch. For example, if there are five VLANs configured on the switch, then there are five separate Spanning Tree instances, each with its own root VLAN. In essence, a VLAN is a virtual bridge in that it will have its own bridge ID and configurable STP parameters, such as protocol, priority, hello time, max age, and forward delay.
- Port state is determined on a per VLAN basis. For example, port connections in VLAN 10 are only examined for redundancy within VLAN 10 across all switches. If a port in VLAN 10 and a port in VLAN 20 both connect to the same switch within their respective VLANs, they are not considered redundant data paths and STP will not block one of them. However, if two ports within VLAN 10 both connect to the same switch, then STP will transition one of these ports to a blocking state.
- Fixed (untagged) ports participate in the single Spanning Tree instance that applies to their configured default VLAN.
- 802.1Q tagged ports participate in an 802.1Q Spanning Tree instance that allows the Spanning Tree to extend across tagged VLANs. As a result, a tagged port may participate in more than one Spanning Tree instance; one for each VLAN that the port carries.
- If a VLAN contains both fixed and tagged ports, then a hybrid of the two Spanning Tree instances (single and 802.1Q) is applied. If a VLAN appears as a tag on a port, then the BPDU for that VLAN are also tagged. However, if a VLAN appears as the configured default VLAN for the port, then BPDU are not tagged and the single Spanning Tree instance applies.

To change the Spanning Tree operating mode to 1x1, enter the following command:

```
-> bridge mode 1x1
```

The following diagram shows a switch running in the 1x1 Spanning Tree mode and shows Spanning Tree participation for both fixed and tagged ports.



1x1 (single and 802.1Q) Spanning Tree Example

In the above example, STP2 is a single Spanning Tree instance since VLAN 5 contains only fixed ports. STP 3 and STP 4 are a combination of single and 802.1Q Spanning Tree instances because VLAN 2 contains both fixed and tagged ports. On ports where VLAN 2 is the default VLAN, BPDU are not tagged. On ports where VLAN 2 is a tagged VLAN, BPDU are also tagged.

Using 1x1 Spanning Tree Mode with PVST+

In order to interoperate with Cisco's proprietary Per Vlan Spanning Tree (PVST+) mode, the current Alcatel-Lucent 1x1 Spanning Tree mode allows OmniSwitch ports to transmit and receive either the standard IEEE BPDUs or Cisco's proprietary PVST+ BPDUs. When PVST+ mode is enabled, a user port operates in 1x1 mode initially by default, until it detects a PVST+ BPDU which will enable that port to operate in the Cisco PVST+ compatible mode automatically. Thus, an OmniSwitch can have ports running in 1x1 mode when connecting to another OmniSwitch, or ports running in Cisco PVST+ mode when connecting to a Cisco switch. So both the Alcatel-Lucent 1x1 and Cisco PVST+ modes can co-exist on the same OmniSwitch and yet interoperate correctly with a Cisco switch using the standard Spanning Tree protocols (802.1d or 802.1w). Note that in the flat Spanning Tree mode, both the OmniSwitch and Cisco switches can interoperate seamlessly using the standard MSTP protocol.

OmniSwitch PVST+ Interoperability

Native VLAN and OmniSwitch Default VLAN

Cisco uses the standard IEEE BPDU format for the native VLAN (i.e., VLAN 1 by default) over an 802.1Q trunk. Thus, by default the Common Spanning Tree (CST) instance of the native VLAN 1 for all Cisco switches and the STP instance for a port's default VLAN on an OmniSwitch will interoperate and successfully create a loop-free topology.

802.1q Tagged VLANs

For 802.1q tagged VLANs, Cisco uses a proprietary frame format which differs from the standard IEEE BPDU format used by Alcatel-Lucent 1X1 mode, thus preventing Spanning Tree topologies for tagged vlans from interoperating over the 802.1Q trunk.

In order to interoperate with Cisco PVST+ mode, the current Alcatel-Lucent *1x1* mode has an option to recognize Cisco's proprietary PVST+ BPDUs and allow any user port on an OmniSwitch to send and receive PVST+ BPDUs, so that loop-free topologies for the tagged VLANs can be created between OmniSwitch and Cisco switches.

Configuration Overview

You can use the **bridge mode 1X1 pvst+** command to globally enable the PVST+ interoperability mode on an OmniSwitch:

```
-> bridge mode 1x1 pvst+ enable
```

To disable the PVST+ mode interoperability mode on an OmniSwitch, use the following command:

```
-> bridge mode 1x1 pvst+ disable
```

The **bridge port pvst+** command is used to configure how a particular port will handle BPDUs when connecting to a Cisco switch.

You can use the **bridge port pvst+** command with the enable option to configure the port to handle only the PVST+ BPDUs and IEEE BPDUs for VLAN 1 (Cisco native VLAN for CST). For example:

```
-> bridge port 1/3 pvst+ enable
```

The following will cause a port to exit from the enable state:

- When the link status of the port changes.
- When the administrative status of the port changes.
- When the PVST+ status of the port is changed to disable or auto.

You can use the **bridge port pvst+** command with the disable option to configure the port to handle only IEEE BPDUs and to drop all PVST+ BPDUs. For example:

```
-> bridge port 1/3 pvst+ disable
```

You can use the **bridge port pvst+** command with the auto option to configure the port to handle IEEE BPDUs initially (i.e., disable state). Once a PVST+ BPDU is received, it will then handle PVST+ BPDUs and IEEE BPDUs for a Cisco native VLAN. For example:

```
-> bridge port 1/3 pvst+ auto
```

Note. By default, a port is configured for PVST+ auto mode on an Omniswitch.

The following show command displays the PVST+ status.

```
-> show spantree mode

Spanning Tree Global Parameters
Current Running Mode   : 1x1,
Current Protocol       : N/A (Per VLAN),
```

```
Path Cost Mode       : 32 BIT,  
Auto Vlan Containment : N/A  
Cisco PVST+ mode    : Enabled
```

BPDU Processing in PVST+ Mode

A port on an OmniSwitch operating in PVST+ mode will process BPDUs as follows:

If the default VLAN of a port is VLAN 1 then:

- Send and receive IEEE untagged BPDUs for VLAN 1
- Don't send and receive PVST+ tagged BPDUs for VLAN 1
- Send and receive tagged PVST+ BPDUs for other tagged VLANs.

If the default VLAN of a port is not VLAN 1 then:

- Send and receive IEEE untagged BPDUs for VLAN 1
- Don't send and receive PVST+ tagged BPDUs for VLAN 1
- Send and receive untagged PVST+ BPDUs for the port's default VLAN
- Send and receive tagged PVST+ BPDUs for other tagged VLANs

Recommendations and Requirements for PVST+ Configurations

- It is mandatory that all the Cisco switches have the Mac Reduction Mode feature enabled in order to interoperate with an OmniSwitch in PVST+ mode. This will avoid any unexpected election of a root bridge.
- You can assign the priority value only in the multiples of 4096 to be compatible with the Cisco MAC Reduction mode; any other values will result in an error message. Also, the existing 1x1 priority values will be restored when changing from PVST+ mode back to 1x1 mode. For more information on priority, refer [“Configuring the Bridge Priority” on page 11-20](#).
- In a mixed OmniSwitch and Cisco environment, it is highly recommended to enable PVST+ mode on all OmniSwitches in order to maintain the same root bridge for the topology. It is possible that the new root bridge might be elected as a result of inconsistencies of MAC reduction mode when connecting an OmniSwitch that does not support Cisco PVST+ mode to an OmniSwitch with the PVST+ mode enabled. In this case, the root bridge priority must be changed manually to maintain the same root bridge. For more information on priority, refer [“Configuring the Bridge Priority” on page 11-20](#).
- A Cisco switch running in PVST mode (another Cisco proprietary mode prior to 802.1q standard) is not compatible with an OmniSwitch running in 1X1 PVST+ mode.
- Both Cisco and an OmniSwitch support two default path cost modes; long or short. It is recommended that the same default path cost mode be configured in the same way on all switches so that the path costs for similar interface types will be consistent when connecting ports between OmniSwitch and Cisco Switches. For more information on path cost mode, refer [“Configuring the Path Cost Mode” on page 11-24](#).
- Dynamic aggregate link (LACP) functions properly between OmniSwitch and Cisco switches. The Cisco switches send the BPDUs only on one physical link of the aggregate, similar to the OmniSwitch Primary port functionality. The path cost assigned to the aggregate link is not the same between OmniSwitch and Cisco switches since vendor-specific formulas are used to derive the path cost. Manual configuration is recommended to match the Cisco path cost assignment for an aggregate link.

For more information on the configuration of path cost for aggregate links, refer [“Path Cost for Link Aggregate Ports”](#) on page 11-32.

The table below shows the default Spanning Tree values.

Parameters	OmniSwitch	Cisco
Mac Reduction Mode	Enabled	Disabled
Bridge Priority	32768	32768
Port Priority	128	32 (catOS) / 128 (IOS)
Port Path Cost	IEEE Port Speed Table	IEEE Port Speed Table
Aggregate Path Cost	Proprietary Table	Avg Path Cost / NumPorts
Default Path Cost Mode	Short (16-bit)	Short (16-bit)
Max Age	20	20
Hello Time	2	2
Forward Delay Time	15	15
Default Protocol	RSTP (1w) Per Vlan	PVST+ (1d) Per Switch

Configuring STP Bridge Parameters

The Spanning Tree software is active on all switches by default and uses default bridge and port parameter values to calculate a loop free topology. It is only necessary to configure these parameter values if it is necessary to change how the topology is calculated and maintained.

Note the following when configuring Spanning Tree bridge parameters:

- When a switch is running in the 1x1 Spanning Tree mode, each VLAN is in essence a virtual bridge with its own Spanning Tree instance and configurable bridge parameters.
- When the switch is running in the flat mode and STP (802.1D) or RSTP (802.1w) is the active protocol, bridge parameter values are only configured for the flat mode instance.
- If MSTP (802.1s) is the active protocol, then the priority value is configurable for each Multiple Spanning Tree Instance (MSTI). All other parameters, however, are still only configured for the flat mode instance and are applied across all MSTIs.
- Bridge parameter values for a VLAN instance are not active unless Spanning Tree is enabled on the VLAN and at least one active port is assigned to the VLAN. Use the **vlan stp** command to enable or disable a VLAN Spanning Tree instance.
- If Spanning Tree is disabled on a VLAN, active ports associated with that VLAN are excluded from Spanning Tree calculations and will remain in a forwarding state.
- Note that when a switch is running in the flat mode, disabling Spanning Tree on VLAN 1 disables the instance for all VLANs and all active ports are then excluded from any Spanning Tree calculations and will remain in a forwarding state.

To view current Spanning Tree bridge parameter values, use the **bridge rrstp ring vlan-tag** command. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Bridge Configuration Commands Overview

Spanning Tree bridge commands are available in an implicit form and an explicit form. Implicit commands resemble commands that were previously released with this feature. The type of instance configured with these commands is determined by the Spanning Tree operating mode that is active at the time the command is used. For example, if the 1x1 mode is active, the instance number specified with the command implies a VLAN ID. If the flat mode is active, the single flat mode instance is implied and thus configured by the command.

Explicit commands introduce three new keywords: **cist**, **1x1**, and **msti**. Each of these keywords when used with a bridge command explicitly identify the type of instance that the command will configure. As a result, explicit commands only configure the type of instance identified by the explicit keyword, regardless of which mode (1x1 or flat) is active.

The **cist** keyword specifies the Common and Internal Spanning Tree (CIST) instance. The CIST is the single Spanning Tree flat mode instance that is available on all switches. When using STP or RSTP, the CIST is also known as instance 1 or bridge 1. When using MSTP (802.1s), the CIST is also known as instance 0. In either case, an instance number is not required with **cist** commands, as there is only one CIST instance.

The **1x1** keyword indicates that the instance number specified with the command is a VLAN ID. The **msti** keyword indicates that the instance number specified with the command is an 802.1s Multiple Spanning Tree Instance (MSTI).

Note that explicit commands using the **cist** and **msti** keywords are required to define an MSTP (802.1s) configuration. Implicit commands are only allowed for defining STP or RSTP configurations. See [Chapter 10, “Using 802.1Q 2005 Multiple Spanning Tree,”](#) for more information about these keywords and using implicit and explicit commands.

The following is a summary of Spanning Tree bridge configuration commands. For more information about these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Commands	Type	Used for ...
bridge protocol	Implicit	Configuring the protocol for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist protocol	Explicit	Configuring the protocol for the single flat mode instance.
bridge 1x1 protocol	Explicit	Configuring the protocol for a VLAN instance.
bridge priority	Implicit	Configuring the priority value for a VLAN instance or the flat mode instance.
bridge cist priority	Explicit	Configuring the priority value for the single flat mode instance.
bridge msti priority	Explicit	Configuring the protocol for an 802.1s Multiple Spanning Tree Instance (MSTI).
bridge 1x1 priority	Explicit	Configuring the priority value for a VLAN instance.
bridge hello time	Implicit	Configuring the hello time value for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist hello time	Explicit	Configuring the hello time value for the single flat mode instance.

Commands	Type	Used for ...
bridge 1x1 hello time	Explicit	Configuring the hello time value for a VLAN instance.
bridge max age	Implicit	Configuring the maximum age time value for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist max age	Explicit	Configuring the maximum age time value for the single flat mode instance.
bridge 1x1 max age	Explicit	Configuring the maximum age time value for a VLAN instance.
bridge forward delay	Implicit	Configuring the forward delay time value for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist forward delay	Explicit	Configuring the forward delay time value for the single flat mode instance.
bridge 1x1 forward delay	Explicit	Configuring the forward delay time value for a VLAN instance.
bridge bpdu-switching	N/A	Configuring the BPDU switching status for a VLAN.
bridge path cost mode	N/A	Configuring the automatic selection of a 16-bit path cost for STP/RSTP ports and a 32-bit path cost for MSTP ports or sets all path costs to use a 32-bit value.
bridge auto-vlan-containment	N/A	Enables or disables Auto VLAN Containment (AVC) for 802.1s instances.
bridge mode 1x1 pvst+	N/A	Enables or disables PVST+ mode on the switch.

Note. When a snapshot is taken of the switch configuration, the explicit form of all Spanning Tree commands is captured. For example, if the bridge protocol for the flat mode instance was changed from STP to MSTP, then **bridge cist protocol mstp** is the command syntax captured to reflect this in the snapshot file. In addition, explicit commands are captured for both flat and 1x1 mode configurations.

The following sections provide information and procedures for using implicit bridge configuration commands and also includes explicit command examples.

Selecting the Bridge Protocol

The switch supports four Spanning Tree protocols: STP, RSTP, MSTP, and RRSTP (the default). To configure the Spanning Tree protocol for a VLAN instance when the switch is running in the 1x1 mode, enter **bridge** followed by an existing VLAN ID, then **protocol** followed by **stp** or **rstp**. For example, the following command changes the protocol to RSTP for VLAN 455:

```
-> bridge 455 protocol rstp
```

Note that when configuring the protocol value for a VLAN instance, MSTP is not an available option. This protocol is only supported on the flat mode instance.

In addition, the explicit **bridge 1x1 protocol** command configures the protocol for a VLAN instance regardless of which mode (1x1 or flat) is active on the switch. For example, the following command also changes the protocol for VLAN 455 to RSTP:

```
-> bridge 1x1 455 protocol rstp
```

To configure the protocol for the single flat mode instance when the switch is running in either mode (1x1 or flat), use the **bridge protocol** command but do *not* specify an instance number. This command configures the flat mode instance by default, so an instance number is not needed, as shown in the following example:

```
-> bridge protocol mstp
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge protocol** command by specifying **1** as the instance number (e.g., **bridge 1 protocol rstp**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

In addition, the explicit **bridge cist protocol** command configures the protocol for the flat mode instance regardless of which mode (1x1 or flat) is active on the switch. For example, the following command selects the RSTP protocol for the flat mode instance:

```
-> bridge cist protocol mstp
```

Configuring the Bridge Priority

A bridge is identified within the Spanning Tree by its bridge ID (an eight byte hex number). The first two bytes of the bridge ID contain a priority value and the remaining six bytes contain a bridge MAC address.

The bridge priority is used to determine which bridge will serve as the root of the Spanning Tree. The lower the priority value, the higher the priority. If more than one bridge have the same priority, then the bridge with the lowest MAC address becomes the root.

Note. Configuring a Spanning Tree bridge instance with a priority value that will cause the instance to become the root is recommended, instead of relying on the comparison of switch base MAC addresses to determine the root.

If the switch is running in the 1x1 Spanning Tree mode, then a priority value is assigned to each VLAN instance. If the switch is running in the flat Spanning Tree mode, the priority is assigned to the flat mode instance or a Multiple Spanning Tree Instance (MSTI). In both cases, the default priority value assigned is 32768. Note that priority values for an MSTI must be multiples of 4096.

To change the bridge priority value for a VLAN instance, specify a VLAN ID with the **bridge priority** command when the switch is running in the 1x1 mode. For example, the following command changes the priority for VLAN 455 to 25590:

```
-> bridge 455 priority 25590
```

The explicit **bridge 1x1 priority** command configures the priority for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 455 priority 25590
```

Note. If PVST+ mode is enabled on the switch, then the priority values can be assigned only in the multiples of 4096 to be compatible with the Cisco MAC Reduction mode; any other values will result in an error message.

To change the bridge priority value for the flat mode instance, use either the **bridge priority** command or the **bridge cist priority** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands change the priority value for the flat mode instance to 12288:

```
-> bridge priority 12288
-> bridge cist priority 12288
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge protocol** command by specifying **1** as the instance number (e.g., **bridge 1 protocol rstp**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

The bridge priority value is also configurable for a Multiple Spanning Tree Instance (MSTI). To configure this value for an MSTI, use the explicit **bridge msti priority** command and specify the MSTI ID for the instance number and a priority value that is a multiple of 4096. For example, the following command configures the priority value for MSTI 10 to 61440:

```
-> bridge msti 10 priority 61440
```

Note that when MSTP is the active flat mode protocol, explicit Spanning Tree bridge commands are required to configure parameter values. Implicit commands are for configuring parameters when the STP or RSTP protocols are in use. See [Chapter 10, “Using 802.1Q 2005 Multiple Spanning Tree,”](#) for more information.

Configuring the Bridge Hello Time

The bridge hello time interval is the number of seconds a bridge will wait between transmissions of Configuration BPDU. When a bridge is attempting to become the root or if it has become the root or a designated bridge, it sends Configuration BPDU out all forwarding ports once every hello time value.

The hello time propagated in a root bridge Configuration BPDU is the value used by all other bridges in the tree for their own hello time. Therefore, if this value is changed for the root bridge, all other bridges associated with the same STP instance will adopt this value as well.

Note that lowering the hello time interval improves the robustness of the Spanning Tree algorithm. Increasing the hello time interval lowers the overhead of Spanning Tree processing.

If the switch is running in the 1x1 Spanning Tree mode, then a hello time value is defined for each VLAN instance. If the switch is running in the flat Spanning Tree mode, then a hello time value is defined for the single flat mode instance. In both cases, the default hello time value used is 2 seconds.

To change the bridge hello time value for a VLAN instance, specify a VLAN ID with the **bridge hello time** command when the switch is running in the 1x1 mode. For example, the following command changes the hello time for VLAN 455 to 5 seconds:

```
-> bridge 455 hello time 5
```

The explicit **bridge 1x1 hello time** command configures the hello time value for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 455 hello time 5
```

To change the bridge hello time value for the flat mode instance, use either the **bridge hello time** command or the **bridge cist hello time** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands change the hello time value for the flat mode instance to 12288:

```
-> bridge hello time 10  
-> bridge cist hello time 10
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge hello time** command by specifying **1** as the instance number (e.g., **bridge 1 hello time 5**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

Note that the bridge hello time is not configurable for Multiple Spanning Tree Instances (MSTI). These instances inherit the hello time from the flat mode instance (CIST).

Configuring the Bridge Max Age Time

The bridge max age time specifies how long, in seconds, the bridge retains Spanning Tree information it receives from Configuration BPDU. When a bridge receives a BPDU, it updates its configuration information and the max age timer is reset. If the max age timer expires before the next BPDU is received, the bridge will attempt to become the root, designated bridge, or change its root port.

The max age time propagated in a root bridge Configuration BPDU is the value used by all other bridges in the tree for their own max age time. Therefore, if this value is changed for the root bridge, all other VLANs associated with the same instance will adopt this value as well.

If the switch is running in the 1x1 Spanning Tree mode, then a max age time value is defined for each VLAN instance. If the switch is running in the flat Spanning Tree mode, then the max age value is defined for the flat mode instance. In both cases, the default max age time used is 20 seconds.

Note that configuring a low max age time may cause Spanning Tree to reconfigure the topology more often.

To change the bridge max age time value for a VLAN instance, specify a VLAN ID with the **bridge max age** command when the switch is running in the 1x1 mode. For example, the following command changes the max age time for VLAN 455 to 10 seconds:

```
-> bridge 455 max age 10
```


The explicit **bridge 1x1 max age** command configures the max age time for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 455 max age 10
```

To change the max age time value for the flat mode instance, use either the **bridge max age** command or the **bridge cist max age** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands change the max age time for the flat mode instance to 10:

```
-> bridge max age 10
-> bridge cist max age 10
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge max age** command by specifying **1** as the instance number (e.g., **bridge 1 max age 30**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

Note that the max age time is not configurable for Multiple Spanning Tree Instances (MSTI). These instances inherit the max age time from the flat mode instance (CIST).

Configuring the Bridge Forward Delay Time

The bridge forward delay time specifies how long, in seconds, a port remains in the learning state while it is transitioning to a forwarding state. In addition, when a topology change occurs, the forward delay time value is used to age out all dynamically learned addresses in the MAC address forwarding table. For more information about the MAC address table, see [Chapter 2, “Managing Source Learning.”](#)

The forward delay time propagated in a root bridge Configuration BPDU is the value used by all other bridges in the tree for their own forward delay time. Therefore, if this value is changed for the root bridge, all other bridges associated with the same instance will adopt this value as well.

If the switch is running in the 1x1 Spanning Tree mode, then a forward delay time value is defined for each VLAN instance. If the switch is running in the flat Spanning Tree mode, then the forward delay time value is defined for the flat mode instance. In both cases, the default forward delay time used is 15 seconds.

Note that specifying a low forward delay time may cause temporary network loops, because packets may get forwarded before Spanning Tree configuration or change notices have reached all nodes in the network.

To change the bridge forward delay time value for a VLAN instance, specify a VLAN ID with the **bridge forward delay** command when the switch is running in the 1x1 mode. For example, the following command changes the forward delay time for VLAN 455 to 10 seconds:

```
> bridge 455 forward delay 20
```

The explicit **bridge 1x1 forward delay** command configures the forward delay time for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 455 forward delay 20
```

To change the forward delay time value for the flat mode instance, use either the **bridge forward delay** command or the **bridge cist forward delay** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands change the forward delay time for the flat mode instance to 10:

```
-> bridge forward delay 10
-> bridge cist forward delay 10
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge forward delay** command by specifying **1** as the instance number (e.g., **bridge 1 forward delay 30**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

Note that the forward delay time is not configurable for Multiple Spanning Tree Instances (MSTI). These instances inherit the forward delay time from the flat mode instance (CIST).

Enabling/Disabling the VLAN BPDU Switching Status

By default, BPDU are not switched on ports associated with VLANs that have Spanning Tree disabled. This may result in a network loop if the VLAN has redundant paths to one or more other switches. Allowing VLANs that have Spanning Tree disabled to forward BPDU to all ports in the VLAN, can help to avoid this problem.

To enable or disable BPDU switching on a VLAN, enter **bridge** followed by an existing VLAN ID (or VLAN 1 if using a flat Spanning Tree instance) then **bpdu-switching** followed by **enable** or **disable**. For example, the following commands enable BPDU switching on VLAN 10 and disable it on VLAN 20:

```
-> bridge 10 bpdu-switching enable
-> bridge 20 bpdu-switching disable
```

Note. Make sure that disabling BPDU switching on a Spanning Tree disabled VLAN will not cause network loops to go undetected.

Configuring the Path Cost Mode

The path cost mode controls whether the switch uses a 16-bit port path cost (PPC) or a 32-bit PPC. When a 32-bit PPC switch connects to a 16-bit PPC switch, the 32-bit switch will have a higher PPC value that will advertise an inferior path cost to the 16-bit switch. In this case, it may be desirable to set the 32-bit switch to use STP or RSTP with a 16-bit PPC value.

By default, the path cost mode is set to automatically use a 16-bit value for all ports that are associated with an STP instance or an RSTP instance and a 32-bit value for all ports associated with an MSTP value. It is also possible to set the path cost mode to always use a 32-bit regardless of which protocol is active.

To change the path cost mode, use the **bridge path cost mode** command and specify either **auto** (uses PPC value based on protocol) or **32bit**. (always use a 32-bit PPC value). For example, the following command changes the default path cost mode, which is automatic, to 32-bit mode:

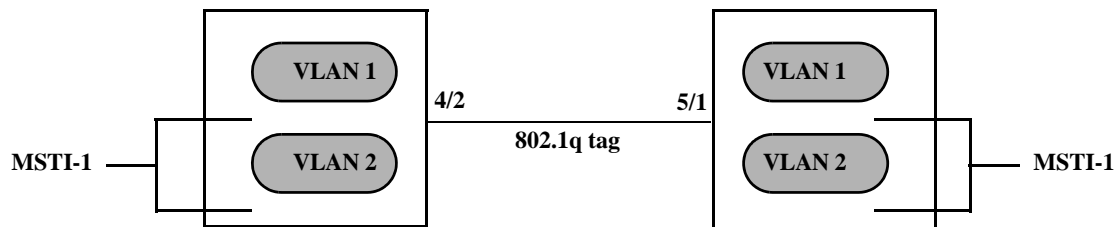
```
-> bridge path cost mode 32bit
```

Note. Cisco supports two default path cost modes: long or short just like in OmniSwitch 1x1 implementation. If you have configured PVST+ mode in the OmniSwitch, it is recommended that the same default path cost mode should be configured in the same way in all the switches, so that, the path costs for similar interface types will be consistent when connecting ports between OmniSwitch and Cisco Switches.

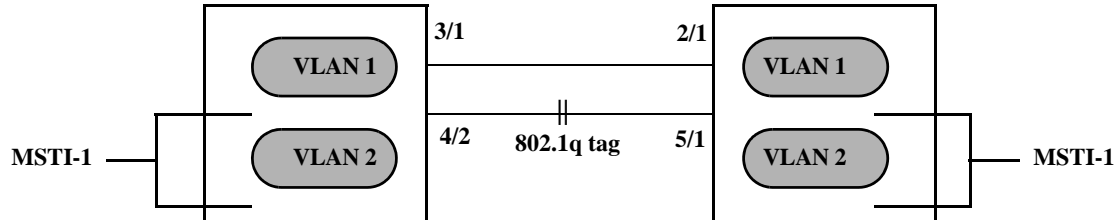
Using Automatic VLAN Containmentment

In a Multiple Spanning Tree (MST) configuration, it is possible for a port that belongs to a VLAN that is not a member of an instance to become the root port for that instance. This can cause a topology change that could lead to a loss of connectivity between VLANs/switches. Enabling Automatic VLAN Containmentment (AVC) helps to prevent this from happening by making such a port an undesirable choice for the root.

When AVC is enabled, it identifies undesirable ports and automatically configures them with an infinite path cost value. For example, in the following diagram a link exists between VLAN 2 on two different switches. The ports that provide this link belong to default VLAN 1 but are tagged with VLAN 2. In addition, VLAN 2 is mapped to MSTI 1 on both switches.



In the above diagram, port 4/2 is the Root port and port 5/1 is a Designated port for MSTI 1. AVC is not enabled. If another link with the same speed and lower port numbers is added to default VLAN 1 on both switches, the new link becomes the root for MSTI 1 and the tagged link between VLAN 2 is blocked, as shown below:



If AVC was enabled in the above example, AVC would have assigned the new link an infinite path cost value that would make this link undesirable as the root for MSTI 1.

Balancing VLANs across links according to their Multiple Spanning Tree Instance (MSTI) grouping is highly recommended to ensure that there is not a loss of connectivity during any possible topology changes. Enabling AVC on the switch is another way to prevent undesirable ports from becoming the root for an MSTI.

By default AVC is disabled on the switch. Use the **bridge auto-vlan-containmentment** command to globally enable this feature for all MSTIs. Once AVC is globally enabled, then it is possible to disable AVC for individual MSTIs using the same command. For example, the following commands globally enable AVC and then disable it for MSTI 10:

```
-> bridge auto-vlan-containmentment enable
-> bridge msti 10 auto-vlan-containmentment disable
```

Note that an administratively set port path cost takes precedence and prevents AVC configuration of the path cost. The exception to this is if the port path cost is administratively set to zero, which resets the path cost to the default value. In addition, AVC does not have any effect on root bridges.

Configuring STP Port Parameters

The following sections provide information and procedures for using CLI commands to configure STP port parameters. These parameters determine the behavior of a port for a specific Spanning Tree instance.

When a switch is running in the 1x1 STP mode, each VLAN is in essence a virtual STP bridge with its own STP instance and configurable parameters. To change STP port parameters while running in this mode, a VLAN ID is specified to identify the VLAN STP instance associated with the specified port. When a switch is running in the flat Spanning Tree mode, VLAN 1 is specified for the VLAN ID.

Only bridged ports participate in the Spanning Tree Algorithm. A port is considered bridged if it meets all the following criteria:

- Port is either a fixed (non-mobile) port, an 802.1Q tagged port, or a link aggregate logical port.
- Spanning tree is enabled on the port.
- Port is assigned to a VLAN that has Spanning Tree enabled.
- Port state (forwarding or blocking) is dynamically determined by the Spanning Tree Algorithm, not manually set.

Bridge Configuration Commands Overview

Spanning Tree port commands are available in an implicit form and an explicit form. Implicit commands resemble commands that were previously released with this feature. The type of instance configured with these commands is determined by the Spanning Tree operating mode that is active at the time the command is used. For example, if the 1x1 mode is active, the instance number specified with the command implies a VLAN ID. If the flat mode is active, the single flat mode instance is implied and thus configured by the command.

Explicit commands introduce three new keywords: **cist**, **1x1**, and **msti**. Each of these keywords when used with a port command explicitly identify the type of instance that the command will configure. As a result, explicit commands only configure the type of instance identified by the explicit keyword regardless of which mode (1x1 or flat) is active.

The **cist** keyword specifies the Common and Internal Spanning Tree (CIST) instance. The CIST is the single Spanning Tree flat mode instance that is available on all switches. When using STP or RSTP, the CIST is also known as instance 1 or bridge 1. When using MSTP, the CIST is also known as instance 0. In either case, an instance number is not required with **cist** commands, as there is only one CIST instance.

The **1x1** keyword indicates that the instance number specified with the command is a VLAN ID. The **msti** keyword indicates that the instance number specified with the command is a Multiple Spanning Tree Instance (MSTI).

Note that explicit commands using the **cist** and **msti** keywords are required to define an MSTP configuration. Implicit commands are only allowed for defining STP or RSTP configurations. See [Chapter 10, "Using 802.1Q 2005 Multiple Spanning Tree,"](#) for more information about these keywords and using implicit and explicit commands.

The following is a summary of Spanning Tree port configuration commands. For more information about these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Commands	Type	Used for ...
bridge slot/port	Implicit	Configuring the port Spanning Tree status for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist slot/port	Explicit	Configuring the port Spanning Tree status for the single flat mode instance.
bridge 1x1 slot/port	Explicit	Configuring the port Spanning Tree status for a VLAN instance.
bridge slot/port priority	Implicit	Configuring the port priority value for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist slot/port priority	Explicit	Configuring the port priority value for the single flat mode instance.
bridge msti slot/port priority	Explicit	Configuring the port priority value for a Multiple Spanning Tree Instance (MSTI).
bridge 1x1 slot/port priority	Explicit	Configuring the port priority value for a VLAN instance.
bridge slot/port path cost	Implicit	Configuring the port path cost value for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist slot/port path cost	Explicit	Configuring the port path cost value for the single flat mode instance.
bridge msti slot/port path cost	Explicit	Configuring the port path cost value for a Multiple Spanning Tree Instance (MSTI).
bridge 1x1 slot/port path cost	Explicit	Configuring the port path cost value for a VLAN instance.
bridge slot/port mode	Explicit	Configuring the port Spanning Tree mode (dynamic or manual) for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist slot/port mode	Implicit	Configuring the port Spanning Tree mode (dynamic or manual) for the single flat mode instance.
bridge 1x1 slot/port mode	Explicit	Configuring the port Spanning Tree mode (dynamic or manual) for a VLAN instance.
bridge slot/port connection	Explicit	Configuring the port connection type for a VLAN instance when the 1x1 mode is active or the single Spanning Tree instance when the flat mode is active.
bridge cist slot/port connection	Implicit	Configuring the port connection type for the single flat mode instance.
bridge 1x1 slot/port connection	Explicit	Configuring the port connection type for a VLAN instance.

Commands	Type	Used for ...
bridge cist slot/port admin-edge	Explicit	Configures the connection type for a port or an aggregate of ports for the flat mode Common and Internal Spanning Tree (CIST).
bridge 1x1 slot/port admin-edge	Explicit	Configures the connection type for a port or an aggregate of ports for a 1x1 mode VLAN instance.
bridge cist slot/port auto-edge	Explicit	Configures a port or an aggregate of ports for the flat mode Common and Internal Spanning Tree (CIST) as an edge port, automatically.
bridge 1x1 slot/port auto-edge	Explicit	Configures a port or an aggregate of ports for the 1x1 mode VLAN instance as an edge port, automatically.
bridge cist slot/port restricted-role	Explicit	Configures the restricted role status for a port or an aggregate of ports for the flat mode Common and Internal Spanning Tree (CIST) as a restricted role port.
bridge 1x1 slot/port restricted-role	Explicit	Configures a port or an aggregate of ports for the 1x1 mode VLAN instance as a restricted role port.
bridge cist slot/port restricted-tcn	Explicit	Configures a port or an aggregate of ports for the flat mode Common and Internal Spanning Tree (CIST) to support the restricted TCN capability.
bridge 1x1 slot/port restricted-tcn	Explicit	Configures a port or an aggregate of ports for the 1x1 mode VLAN instance to support the restricted TCN capability.
bridge cist txholdcount	Explicit	Limits the transmission of BPDU through a given port for the flat mode Common and Internal Spanning Tree (CIST).
bridge 1x1 txholdcount	Explicit	Limits the transmission of BPDU through a given port for the 1x1 mode VLAN instance.
bridge port pvst+	Explicit	Configures the type of BPDU to be used on a port when PVST+ mode is enabled.

The following sections provide information and procedures for using implicit Spanning Tree port configuration commands and also includes explicit command examples.

Note. When a snapshot is taken of the switch configuration, the explicit form of all Spanning Tree commands is captured. For example, if the bridge protocol for the flat mode instance was changed from STP to MSTP, then **bridge cist protocol mstp** is the command syntax captured to reflect this in the snapshot file. In addition, explicit commands are captured for both flat and 1x1 mode configurations.

Enabling/Disabling Spanning Tree on a Port

By default, Spanning Tree is enabled on all ports. When Spanning Tree is disabled on a port, the port is put in a forwarding state for the specified instance. For example, if a port is associated with both VLAN 10 and VLAN 20 and Spanning Tree is disabled on the port for VLAN 20, the port state is set to forwarding for VLAN 20. However, the VLAN 10 instance still controls the port's state as it relates to VLAN 10. This example assumes the switch is running in the 1x1 Spanning Tree mode.

If the switch is running in the flat Spanning Tree mode, then disabling the port Spanning Tree status applies across all VLANs associated with the port. The flat mode instance is specified as the port's instance, even if the port is associated with multiple VLANs.

To change the port Spanning Tree status for a VLAN instance, specify a VLAN ID with the **bridge slot/port** command when the switch is running in the 1x1 mode. For example, the following commands enable Spanning Tree on port 8/1 for VLAN 10 and disable STP on port 6/2 for VLAN 20:

```
-> bridge 10 8/1 enable
-> bridge 20 6/2 disable
```

The explicit **bridge 1x1 slot/port** command configures the priority for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following commands perform the same function as the commands in the previous example:

```
-> bridge 1x1 10 8/1 enable
-> bridge 1x1 20 6/2 disable
```

To change the port Spanning Tree status for the flat mode instance, use either the **bridge slot/port** command or the **bridge cist slot/port** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands disable the Spanning Tree status on port 1/24 for the flat mode instance:

```
-> bridge 1/24 disable
-> bridge cist 1/24 disable
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge slot/port** command by specifying **1** as the instance number (e.g., **bridge 1 1/24 enable**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

Spanning Tree on Link Aggregate Ports

Physical ports that belong to a link aggregate do not participate in the Spanning Tree Algorithm. Instead, the algorithm is applied to the aggregate logical link (virtual port) that represents a collection of physical ports.

To enable or disable the Spanning Tree status for a link aggregate, use the **bridge slot/port** commands described above but specify a link aggregate control number instead of a slot and port. For example, the following command disables Spanning Tree for link aggregate 10 associated with VLAN 755:

```
-> bridge 755 10 disable
```

For more information about configuring an aggregate of ports, see [Chapter 23, "Configuring Static Link Aggregation,"](#) and [Chapter 24, "Configuring Dynamic Link Aggregation."](#)

Configuring Port Priority

A bridge port is identified within the Spanning Tree by its Port ID (a 16-bit or 32-bit hex number). The first 4 bits of the Port ID contain a priority value and the remaining 12 bits contain the physical switch port number. The port priority is used to determine which port offers the best path to the root when multiple paths have the same path cost. The port with the highest priority (lowest numerical priority value) is selected and the others are put into a blocking state. If the priority values are the same for all ports in the path, then the port with the lowest physical switch port number is selected.

By default, Spanning Tree is enabled on a port and the port priority value is set to 7. If the switch is running in the 1x1 Spanning Tree mode, then the port priority applies to the specified VLAN instance associated with the port. If the switch is running in the flat Spanning Tree mode, then the port priority applies across all VLANs associated with the port. The flat mode instance is specified as the port's instance, even if the port is associated with multiple VLANs.

To change the port priority value for a VLAN instance, specify a VLAN ID with the **bridge slot/port priority** command when the switch is running in the 1x1 mode. For example, the following command sets the priority value for port 8/1 to 3 for the VLAN 10 instance:

```
-> bridge 10 8/1 priority 3
```

The explicit **bridge cist slot/port priority** command configures the port priority value for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 10 8/1 priority 3
```

To change the port priority value for the flat mode instance, use either the **bridge slot/port priority** command or the **bridge cist slot/port priority** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands change the priority value for port 1/24 for the flat mode instance to 15:

```
-> bridge 1/24 priority 15  
-> bridge cist 1/24 priority 10
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge slot/port priority** command by specifying **1** as the instance number (e.g., **bridge 1 1/24 priority 15**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

The port priority value is also configurable for a Multiple Spanning Tree Instance (MSTI). To configure this value for an MSTI, use the explicit **bridge msti slot/port priority** command and specify the MSTI ID for the instance number. For example, the following command configures the priority value for port 1/12 for MSTI 10 to 5:

```
-> bridge msti 10 1/12 priority 5
```

Note that when MSTP is the active flat mode protocol, explicit Spanning Tree bridge commands are required to configure parameter values. Implicit commands are for configuring parameters when the STP or RSTP protocols are in use. See [Chapter 10, "Using 802.1Q 2005 Multiple Spanning Tree,"](#) for more information.

Port Priority on Link Aggregate Ports

Physical ports that belong to a link aggregate do not participate in the Spanning Tree Algorithm. Instead, the algorithm is applied to the aggregate logical link (virtual port) that represents a collection of physical ports.

To change the port priority for a link aggregate, use the **bridge slot/port priority** commands described above, but specify a link aggregate control number instead of a slot and port. For example, the following command sets the priority for link aggregate 10 associated with VLAN 755 to 9:

```
-> bridge 755 10 priority 9
```

For more information about configuring an aggregate of ports, see [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Configuring Port Path Cost

The path cost value specifies the contribution of a port to the path cost towards the root bridge that includes the port. The root path cost is the sum of all path costs along this same path and is the value advertised in Configuration BPDU transmitted from active Spanning Tree ports. The lower the cost value, the closer the switch is to the root.

Note that type of path cost value used depends on which path cost mode is active (automatic or 32-bit). If the path cost mode is set to automatic, a 16-bit value is used when STP or RSTP is the active protocol and a 32-bit value is used when MSTP is the active protocol. If the mode is set to 32-bit, then a 32-bit path cost value is used regardless of which protocol is active. See [“Configuring the Path Cost Mode” on page 11-24](#) for more information.

If a 32-bit path cost value is in use and the *path_cost* is set to zero, the following IEEE 802.1Q 2005 recommended default path cost values based on link speed are used:

Link Speed	IEEE 802.1D Recommended Value
10 MB	2,000,000
100 MB	200,000
1 GB	20,000

If a 16-bit path cost value is in use and the *path_cost* is set to zero, the following IEEE 802.1D recommended default path cost values based on link speed are used:

Link Speed	IEEE 802.1D Recommended Value
4 Mbps	250
10 Mbps	100
16 Mbps	62
100 Mbps	19
1 Gbps	4

By default, Spanning Tree is enabled on a port and the path cost is set to zero. If the switch is running in the 1x1 Spanning Tree mode, then the port path cost applies to the specified VLAN instance associated with the port. If the switch is running in the flat Spanning Tree mode, then the port path cost applies across

all VLANs associated with the port. The flat mode instance is specified as the port's instance, even if the port is associated with other VLANs.

To change the port path cost value for a VLAN instance, specify a VLAN ID with the **bridge slot/port path cost** command when the switch is running in the 1x1 mode. For example, the following command configures a 16-bit path cost value for port 8/1 for VLAN 10 to 19 (the port speed is 100 MB, 19 is the recommended value).

```
-> bridge 10 8/1 path cost 19
```

The explicit **bridge 1x1 slot/port path cost** command configures the port path cost value for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 10 8/1 path cost 19
```

To change the port path cost value for the flat mode instance, use either the **bridge slot/port path cost** command or the **bridge cist slot/port path cost** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands configure a 32-bit path cost value for port 1/24 for the flat mode instance to 20,000 (the port speed is 1 GB, 20,000 is the recommended value):

```
-> bridge 1/24 path cost 20000
-> bridge cist 1/24 path cost 20000
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge slot/port path cost** command by specifying **1** as the instance number (e.g., **bridge 1 1/24 path cost 19**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

The port path cost value is also configurable for a Multiple Spanning Tree Instance (MSTI). To configure this value for an MSTI, use the explicit **bridge msti slot/port path cost** command and specify the MSTI ID for the instance number. For example, the following command configures the path cost value for port 1/12 for MSTI 10 to 19:

```
-> bridge msti 10 1/12 path cost 19
```

Note that when MSTP is the active flat mode protocol, explicit Spanning Tree bridge commands are required to configure parameter values. Implicit commands are for configuring parameters when the STP or RSTP protocols are in use. See [Chapter 10, "Using 802.1Q 2005 Multiple Spanning Tree,"](#) for more information.

Path Cost for Link Aggregate Ports

Physical ports that belong to a link aggregate do not participate in the Spanning Tree Algorithm. Instead, the algorithm is applied to the aggregate logical link (virtual port) that represents a collection of physical ports. By default, Spanning Tree is enabled on the aggregate logical link and the path cost value is set to zero.

If a 32-bit path cost value is in use and the *path_cost* for a link aggregate is set to zero, the following default values based on link speed and link aggregate size are used:

Link Speed	Aggregate Size (number of links)	Default Path Cost Value
10 MB	2	1,200,000

Link Speed	Aggregate Size (number of links)	Default Path Cost Value
	4	800,000
	8	600,000
100 MB	2	120,000
	4	80,000
	8	60,000
1 GB	2	12,000
	4	8,000
	8	6,000

If a 16-bit path cost value is in use and the *path_cost* for a link aggregate is set to zero, the following default values based on link speed and link aggregate size are used. Note that for Gigabit ports the aggregate size is not applicable in this case:

Link Speed	Aggregate Size (number of links)	Default Path Cost Value
10 Mbps	2	60
	4	40
	8	30
100 Mbps	2	12
	4	9
	8	7
1 Gbps	N/A	3

To change the path cost value for a link aggregate, use the **bridge slot/port path cost** commands described above, but specify a link aggregate control number instead of a slot and port. For example, the following command sets the path cost for link aggregate 10 associated with VLAN 755 to 19:

```
-> bridge 755 10 path cost 19
```

For more information about configuring an aggregate of ports, see [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Configuring Port Mode

There are two port modes supported: manual and dynamic. Manual mode indicates that the port was set by the user to a forwarding or blocking state. The port will operate in the state selected until the state is manually changed again or the port mode is changed to dynamic. Ports operating in a manual mode state do not participate in the Spanning Tree Algorithm. Dynamic mode indicates that the active Spanning Tree Algorithm will determine port state.

By default, Spanning Tree is enabled on the port and the port operates in the dynamic mode. If the switch is running in the 1x1 Spanning Tree mode, then the port mode applies to the specified VLAN instance associated with the port. If the switch is running in the flat Spanning Tree mode, then the port mode applies across all VLANs associated with the port. The flat mode instance is specified as the port's instance, even if the port is associated with other VLANs.

To change the port Spanning Tree mode for a VLAN instance, specify a VLAN ID with the **bridge slot/port mode** command when the switch is running in the 1x1 mode. For example, the following command sets the mode for port 8/1 for VLAN 10 to forwarding.

```
-> bridge 10 8/1 mode forwarding
```

The explicit **bridge 1x1 slot/port mode** command configures the port mode for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 10 8/1 mode forwarding
```

To change the port Spanning Tree mode for the flat mode instance, use either the **bridge slot/port mode** command or the **bridge cist slot/port mode** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands configure the Spanning Tree mode on port 1/24 for the flat mode instance:

```
-> bridge 1/24 mode blocking
-> bridge cist 1/24 mode blocking
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge slot/port mode** command by specifying **1** as the instance number (e.g., **bridge 1 1/24 mode dynamic**). However, this is only available when the switch is already running in the flat mode and STP or RSTP is the active protocol.

Mode for Link Aggregate Ports

Physical ports that belong to a link aggregate do not participate in the Spanning Tree Algorithm. Instead, the algorithm is applied to the aggregate logical link (virtual port) that represents a collection of physical ports. To change the port mode for a link aggregate, use the **bridge slot/port mode** commands described above, but specify a link aggregate control number instead of a slot and port. For example, the following command sets the port mode for link aggregate 10 associated with VLAN 755 to blocking:

```
-> bridge 755 10 mode blocking
```

For more information about configuring an aggregate of ports, see [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Configuring Port Connection Type

Specifying a port connection type is done when using the Rapid Spanning Tree Algorithm and Protocol (RSTP), as defined in the IEEE 802.1w standard. RSTP transitions a port from a blocking state directly to forwarding, bypassing the listening and learning states, to provide a rapid reconfiguration of the Spanning Tree in the event of a path or root bridge failure. Rapid transition of a port state depends on the port's configurable connection type. These types are defined as follows:

- Point-to-point LAN segment (port connects directly to another switch).
- No point-to-point shared media LAN segment (port connects to multiple switches).
- Edge port (port is at the edge of a bridged LAN, does not receive BPDU and has only one MAC address learned). Edge ports, however, will operationally revert to a point to point or a no point to point connection type if a BPDU is received on the port.

A port is considered connected to a point-to-point LAN segment if the port belongs to a link aggregate of ports, or if auto negotiation determines if the port should run in full duplex mode, or if full duplex mode was administratively set. Otherwise, that port is considered connected to a no point-to-point LAN segment.

Rapid transition of a designated port to forwarding can only occur if the port's connection type is defined as a point to point or an edge port. Defining a port's connection type as a point to point or as an edge port makes the port eligible for rapid transition, regardless of what actually connects to the port. However, an alternate port is always allowed to transition to the role of root port regardless of the alternate port's connection type.

Note. Configure ports that will connect to a host (PC, workstation, server, etc.) as edge ports so that these ports will transition directly to a forwarding state and not trigger an unwanted topology change when a device is connected to the port. If a port is configured as a point to point or no point to point connection type, the switch will assume a topology change when this port goes active and will flush and relearn all learned MAC addresses for the port's assigned VLAN.

By default, Spanning Tree is enabled on the port and the connection type is set to auto point to point. The auto point to point setting determines the connection type based on the operational status of the port.

If the switch is running in the 1x1 Spanning Tree mode, then the connection type applies to the specified VLAN instance associated with the port. If the switch is running in the flat Spanning Tree mode, then the connection type applies across all VLANs associated with the port. The flat mode instance is referenced as the port's instance, even if the port is associated with other VLANs.

To change the port connection type for a VLAN instance, specify a VLAN ID with the **bridge slot/port connection** command when the switch is running in the 1x1 mode. For example, the following command defines an edge port connection type for port 8/1 associated with VLAN 10.

```
-> bridge 10 8/1 connection edgeport
```

The explicit **bridge 1x1 slot/port connection** command configures the connection type for a VLAN instance when the switch is running in either mode (1x1 or flat). For example, the following command performs the same function as the command in the previous example:

```
-> bridge 1x1 10 8/1 connection edgeport
```

To change the port Spanning Tree mode for the flat mode instance, use either the **bridge slot/port connection** command or the **bridge cist slot/port connection** command. Note that both commands are available when the switch is running in either mode (1x1 or flat) and an instance number is not required. For example, the following commands configure the connection type for port 1/24 for the flat mode instance:

```
-> bridge 1/24 connection ptp
-> bridge cist 1/24 connection ptp
```

As in previous releases, it is possible to configure the flat mode instance with the **bridge slot/port connection** command by specifying **1** as the instance number (e.g., **bridge 1 1/24 connection noptp**). However, this is only available when the switch is running in the flat mode and STP or RSTP is the active protocol.

Note that the **bridge slot/port connection** command only configures one port at a time.

Connection Type on Link Aggregate Ports

Physical ports that belong to a link aggregate do not participate in the Spanning Tree Algorithm. Instead, the algorithm is applied to the aggregate logical link (virtual port) that represents a collection of physical ports. To change the port connection type for a link aggregate, use the **bridge slot/port connection** commands described above, but specify a link aggregate control number instead of a slot and port. For example, the following command defines link aggregate 1, associated with VLAN 755, as an edge port:

```
-> bridge 755 10 connection edgeport
```

For more information about configuring an aggregate of ports, see [Chapter 23, “Configuring Static Link Aggregation,”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Configuring Edge Port

By default, **auto-edge** functionality is enabled on the ports which implies that the Spanning Tree automatically determines the operational edge port status of the ports.

The **auto-edge** functionality can be enabled or disabled on a port in the flat mode Common and Internal Spanning Tree (CIST) instance by using the **bridge cist slot/port auto-edge** command. Similarly a port in 1x1 instance can be configured by using the **bridge 1x1 slot/port auto-edge** command.

To disable the **auto-edge** functionality of a port in **CIST** instance, enter the following command:

```
-> bridge cist 8/23 auto-edge disable
```

To enable the **auto-edge** functionality of the port, enter the following command:

```
-> bridge cist 8/23 auto-edge enable
```

The administrative edge port status (**admin-edge**) is used to determine the status of the port when automatic edge port configuration (**auto-edge**) is disabled.

To define the administrative edge port status (**admin-edge**) of a port in a CIST instance, use the **bridge cist slot/port admin-edge** command. Similarly for a port in 1x1 instance, use the **bridge 1x1 slot/port admin-edge** command.

Note. If **auto-edge** is enabled on a port, then the **admin-edge** value is overridden.

To enable the administrative edge port status for a port in CIST mode, enter the following command:

```
-> bridge cist 8/23 admin-edge disable
```

Restricting Port Roles (Root Guard)

By default, all ports are eligible for root port selection. A port in a CIST/MSTI instance or 1x1 instance can be prevented from becoming the root port by restricting the role of the port (also referred to as enabling root guard). This is done using the **bridge cist slot/port restricted-role** command or the **bridge 1x1 slot/port restricted-role** command. For example:

```
-> bridge cist 1/24 restricted-role enable
-> bridge 1x1 100 8/1 restricted-role enable
```

Note that the above commands also provide optional syntax; **restricted-role** or **root-guard**. For example, the following two commands perform the same function:

```
-> bridge 1x1 2/1 restricted-role enable
-> bridge 1x1 2/1 root-guard enable
```

When root guard is enabled for a port, it cannot become the root port, even if it is the most likely candidate for becoming the root port. It will be selected as the alternate port when the root port is selected.

Restricting TCN Propagation

By default, all the ports propagate Topology Change Notifications (TCN) or Topology Changes (TC) to other ports.

A port in CIST instance can be restricted from propagating Topology Change Notification (TCN) using the **bridge cist slot/port restricted-tcn** command. Similarly a port in 1x1 instance can be restricted by using the **bridge 1x1 slot/port restricted-tcn** command.

For example, to restrict the port 2/2 from propagating the received TCNs and TCs to the other ports, enter the following command:

```
-> bridge cist 2/2 restricted-tcn enable
```

Limiting BPDU Transmission

The number of BPDUs to be transmitted per port per second can be limited using the **bridge cist txhold-count** command for a CIST instance or **bridge 1x1 txholdcount** commands for a 1x1 instance.

For example, to limit the number of BPDUs to be transmitted by a port in CIST instance to 5, enter the following command:

```
-> bridge cist txholdcount 5
```

Using RRSTP

The Ring Rapid Spanning Tree Protocol (RRSTP) is complimentary to both the Spanning Tree Protocol (STP) as well as the Multiple Spanning Tree Protocol (MSTP). It is designed to provide faster convergence time when switches are connected point to point in a ring topology. RRSTP can only be configured on an OmniSwitch running in flat mode.

RRSTP reduces convergence time by finding the bridge that hosts the alternate (ALT) port and immediately changing the ALT port state to forwarding without altering the MSTP port state. This process quickly enables the data path. The RRSTP frame travels from the point of failure to the ALT port in both directions. The MAC addresses corresponding to the ports in the ring are flushed to make the data path convergence time much faster than the normal MSTP.

While RRSTP is already reacting to the loss of connectivity, the standard MSTP BPDU carrying the link down information is processed in normal fashion at each hop. When this MSTP BPDU reaches the bridge whose ALT port is now in the "ALT FWD" state, due to RRSTP frame processing, it updates the MSTP state of the two ports in the ring as per the MSTP standard.

The following limitations should be noted when using RRSTP:

- There can be no alternate connections for the same instance between any two switches within an RRSTP ring topology.
- A port on a bridge can only be part of one RRSTP ring at any given instance.
- All bridges, which need to be made part of a ring, can be configured only statically.
- Fast convergence will not occur if an RRSTP frame is lost. However, MSTP convergence will still take place at a later time because there is no way of knowing about the RRSTP frame loss.
- RRSTP convergence may not happen when changes in configuration result in an unstable topology.
- If either of the two ports of the RRSTP ring on a bridge goes down or if one of the bridges in the ring goes down, the RRSTP convergence may not happen. However, MSTP convergence will continue without interruption.
- A single switch can participate in up to 128 RRSTP rings.

Configuring RRSTP

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to configure Ring Rapid Spanning Tree Protocol (RRSTP) on a switch.

When configuring RRSTP parameters, you must perform the following steps:

- 1 Enable RRSTP on your switch.** To enable RRSTP globally on a switch, use the [bridge rrstp](#) command, which is described in [“Enabling and Disabling RRSTP”](#) on page 11-39.
- 2 Create RRSTP ring comprising of two ports.** To create an RRSTP ring comprising of two ports, use the [bridge rrstp ring](#) command, which is described in [“Creating and Removing RRSTP Rings”](#) on page 11-39.

Enabling and Disabling RRSTP

To enable RRSTP switch-wide, use the [bridge rrstp](#) command by entering:

```
-> bridge rrstp
```

To disable RRSTP switch-wide, use the **no** form of the command by entering:

```
-> no bridge rrstp
```

You can display the current RRSTP status at a global level using the [show bridge rrstp configuration](#) command.

```
-> show bridge rrstp configuration
RRSTP Global state is Enabled
```

Creating and Removing RRSTP Rings

By default, an RRSTP ring is disabled on the switch. To create an RRSTP ring comprising of two ports, use the [bridge rrstp ring](#) command by entering:

```
-> bridge rrstp ring 1 port1 1/1 port2 1/3 vlan-tag 10 status enable
```

To modify the vlan-tag associated with the ring, use the [bridge rrstp ring vlan-tag](#) command by entering:

```
-> bridge rrstp ring 1 vlan-tag 20
```

To remove an RRSTP ring comprising of two ports, use the **no** form of the command by entering:

```
-> no bridge rrstp ring 1
```

You can display the information of a specific ring or all the rings on the switch using the [show bridge rrstp ring](#) command, as shown:

```
-> show bridge rrstp ring
  RingId      Vlan-Tag      Ring-Port1      Ring-Port2      Ring Status
-----+-----+-----+-----+-----
      2          1000          1/19            1/10            enabled
      6           20           1/1             1/8            disabled
     128           1           0/1             0/31            enabled
```

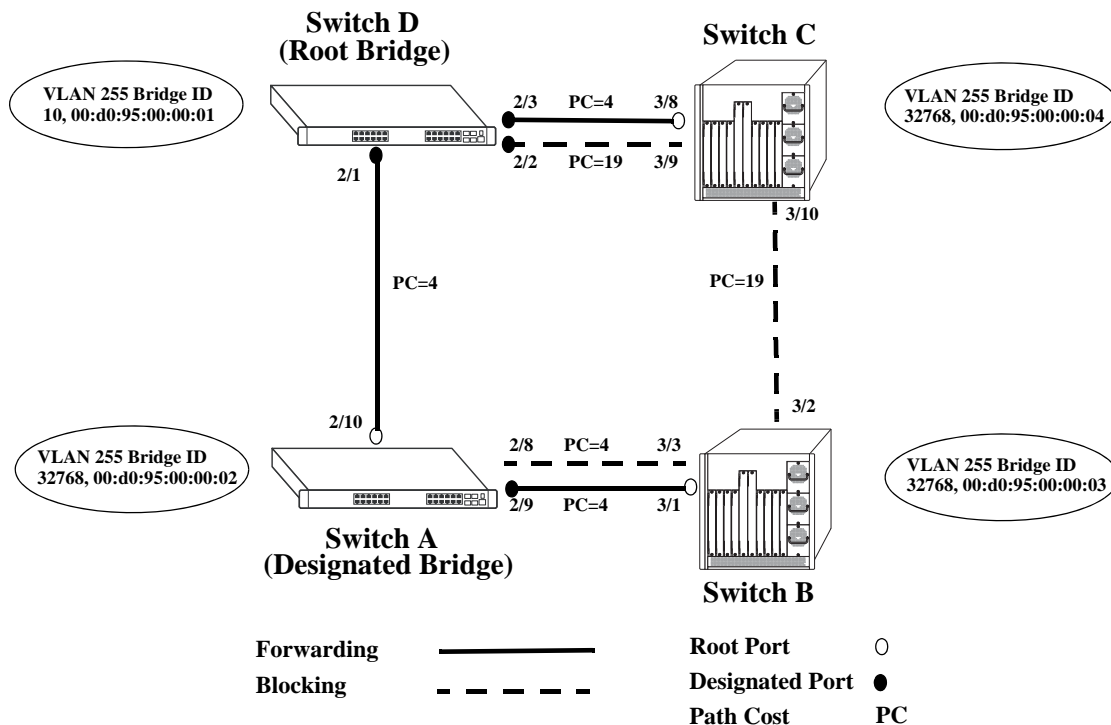
Sample Spanning Tree Configuration

This section provides an example network configuration in which the Spanning Tree Algorithm and Protocol has calculated a loop-free topology. In addition, a tutorial is also included that provides steps on how to configure the example network topology using the Command Line Interface (CLI).

Note that the following example network configuration illustrates using switches operating in the 1x1 Spanning Tree mode and using RSTP (802.1w) to calculate a single data path between VLANs. See [Chapter 10, “Using 802.1Q 2005 Multiple Spanning Tree,”](#) for an overview and examples of using MSTP (802.1s).

Example Network Overview

The following diagram shows a four-switch network configuration with an active Spanning Tree topology, which was calculated based on both configured and default Spanning Tree parameter values:



Example Active Spanning Tree Topology

In the above example topology:

- Each switch is operating in the 1x1 Spanning Tree mode by default.
- Each switch configuration has a VLAN 255 defined. The Spanning Tree administrative status for this VLAN was enabled by default when the VLAN was created.
- VLAN 255 on each switch is configured to use the 802.1w (rapid reconfiguration) Spanning Tree Algorithm and Protocol.
- Ports 2/1-3, 2/8-10, 3/1-3, and 3/8-10 provide connections to other switches and are all assigned to VLAN 255 on their respective switches. The Spanning Tree administrative status for each port is enabled by default.

- The path cost for each port connection defaults to a value based on the link speed. For example, the connection between Switch B and Switch C is a 100 Mbps link, which defaults to a path cost of 19.
- VLAN 255 on Switch D is configured with a Bridge ID priority value of 10, which is less than the same value for VLAN 255 configured on the other switches. As a result, VLAN 255 was elected the Spanning Tree root bridge for the VLAN 255 broadcast domain.
- A root port is identified for VLAN 255 on each switch, except the root VLAN 255 switch. The root port identifies the port that provides the best path to the root VLAN.
- VLAN 255 on Switch A was elected the designated bridge because it offers the best path cost for Switch B to the root VLAN 255 on Switch D.
- Port 2/9 on Switch A is the designated port for the Switch A to Switch B connection because Switch A is the designated bridge for Switch B.
- Redundant connections exist between Switch D and Switch C. Ports 2/2 and 3/9 are in a discarding (blocking) state because this connection has a higher path cost than the connection provided through ports 2/3 and 3/8. As a result, a network loop condition is avoided.
- Redundant connections also exist between Switch A and Switch B. Although the path cost value for both of these connections is the same, ports 2/8 and 3/3 are in a discarding state because their port priority values (not shown) are higher than the same values for ports 2/10 and 3/1.
- The ports that provide the connection between Switch B and Switch C are in a discarding (blocking) state, because this connection has a higher path cost than the other connections leading to the root VLAN 255 on Switch D. As a result, a network loop is avoided.

Example Network Configuration Steps

The following steps provide a quick tutorial that configures the active Spanning Tree network topology shown in the diagram on [page 11-40](#).

- 1** Create VLAN 255 on Switches A, B, C, and D with “Marketing IP Network” for the VLAN description on each switch using the following command:

```
-> vlan 255 name "Marketing IP Network"
```

- 2** Assign the switch ports that provide connections between each switch to VLAN 255. For example, the following commands entered on Switches A, B, C, and D, respectively, assign the ports shown in the example network diagram on [page 11-40](#) to VLAN 255:

```
-> vlan 255 port default 2/8-10
-> vlan 255 port default 3/1-3
-> vlan 255 port default 3/8-10
-> vlan 255 port default 2/1-3
```

- 3** Change the Spanning Tree protocol for VLAN 255 to 802.1w (rapid reconfiguration) on each switch using the following command:

```
-> bridge 255 protocol 1w
```


Verifying the Spanning Tree Configuration

To display information about the Spanning Tree configuration on the switch, use the show commands listed below:

bridge rrstp ring vlan-tag	Displays VLAN Spanning Tree information, including parameter values and topology change statistics.
show spantree ports	Displays Spanning Tree information for switch ports, including parameter values and the current port state.

For more information about the resulting displays from these commands, see the *OmniSwitch 6250 CLI Reference Guide*. An example of the output for the **show spantree** and **show spantree ports** commands is also given in [“Example Network Configuration Steps” on page 11-41](#).

12 Configuring ERP

The ITU-T G.8032/Y.1344 Ethernet Ring Protection (ERP) switching mechanism is a self-configuring algorithm that maintains a loop-free topology while providing data path redundancy and network scalability. ERP provides fast recovery times for Ethernet ring topologies by utilizing traditional Ethernet MAC and bridge functions.

In This Chapter

This chapter provides an overview about how Ethernet Ring Protection (ERP) works and how to configure its parameters through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

The following information and configuration procedures are included in this chapter include:

- [“ERP Overview” on page 12-3.](#)
- [“Interaction With Other Features” on page 12-8.](#)
- [“ERP Configuration Overview and Guidelines” on page 12-11.](#)
- [“Configuring an ERP Ring” on page 12-12.](#)
- [“Sample Ethernet Ring Protection Configuration” on page 12-19.](#)
- [“Verifying the ERP Configuration” on page 12-21.](#)

ERP Specifications

ITU-T G.8032/Y.1344	Ethernet Ring Protection (Hold-off timer not supported) (Non-revertive mode not supported)
ITU-T Y.1731/IEEE 802.1ag	ERP packet compliant with OAM PDU format for CFM
Supported Platforms	OmniSwitch 6250-Metro Models
Maximum number of rings per node	4
Maximum number of rings per ring port	1
Maximum number of nodes per ring	16 (recommended)
Maximum number of ERP protected VLANs per switch.	252 on switch operating in the 1x1 Spanning Tree mode.
Range for ring ID	1 - 2147483647
Range for remote MEPID	1 - 8191
Range for wait-to-restore timer	1 - 12 minutes
Range for guard timer	1 - 200 centi-seconds

ERP Defaults

Parameter Description	Command	Default
ERP ring status	erp-ring	Disabled
RPL status for the node	erp-ring rpl-node	Disabled
The wait-to-restore timer value for the RPL node	erp-ring wait-to-restore	5 minutes
The guard-timer value for the ring node	erp-ring guard-timer	50 centi-seconds
ERP interaction with Ethernet OAM (accept or drop loss of connectivity events from remote endpoint).	erp-ring ethoam-event remote-endpoint	Events are dropped
The NNI-SVLAN association type.	ethernet-service svlan nni	STP

ERP Overview

Ethernet Ring Protection (ERP) is a protection switching mechanism for Ethernet ring topologies, such as multi-ring and ladder networks. This implementation of ERP is based on Recommendation ITU-T G.8032/Y.1344 and uses the ring Automatic Protection Switching (APS) protocol to coordinate the prevention of network loops within a bridged Ethernet ring.

Loop prevention is achieved by allowing the traffic to flow on all but one of the links within the protected Ethernet ring. This link is blocked and is referred to as the Ring Protection Link (RPL). When a ring failure condition occurs, the RPL is unblocked to allow the flow of traffic to continue through the ring.

One designated node within the ring serves as the RPL owner and is responsible for blocking the traffic over the RPL. When a ring failure condition occurs, the RPL owner is responsible for unblocking the RPL so that the link can forward traffic to maintain ring connectivity.

ERP Terms

Ring Protection Link (RPL)—A designated link between two ring nodes that is blocked to prevent a loop on the ring.

RPL Owner—A node connected to an RPL. This node blocks traffic on the RPL during normal ring operations and activates the link to forward traffic when a failure condition occurs on another link in the ring.

Link Monitoring—Ring links are monitored using standard ETH CC OAM messages (CFM). Note that for improved convergence times, this implementation also uses Ethernet link up and link down events.

Signal Fail (SF)—Signal Fail is declared when a failed link or node is detected.

No Request (NR)—No Request is declared when there are no outstanding conditions (e.g., SF) on the node.

Ring APS (R-APS) Messages—Protocol messages defined in Y.1731 and G.8032 that determine the status of the ring.

ERP Service VLAN—Ring-wide VLAN used exclusively for transmission of messages, including R-APS messages.

ERP Protected VLAN—A VLAN that is added to the ERP ring. ERP determines the forwarding state of protected VLANs.

ERP Timers

Wait To Restore (WTR) Timer. To prevent link flapping, this timer is used by the RPL to verify that the ring has stabilized. This timer determines the number of minutes the RPL switch waits before returning the RPL ports to a blocked state after the ring has recovered from a link failure.

Some important points about the WTR Timer are as follows:

- The timer is started when the RPL node receives an R-APS (NR) message that indicates ring protection is no longer required.
- The timer is stopped when the RPL owner receives an R-APS (SF) message while WTR is running, which indicates that an error still exists in the ring.
- When the time runs out, the RPL port is blocked and an R-APS (NR, RB) message is transmitted from both the ring ports to indicate that the RPL is blocked.

- Refer to the “[ERP Specifications](#)” on page 12-2 for timer defaults and valid ranges.

Guard Timer. When the failed link recovers, a ring node will start the Guard Timer. The Guard Timer is used to prevent the ring nodes from receiving outdated R-APS messages that are no longer relevant.

Some important points about the Guard Timer are as follows:

- When the Guard Timer is running, any R-APS messages received are not forwarded.
- The Guard Timer value should be greater than the maximum expected forwarding delay time for which it takes one R-APS message to circulate around the ring. This calculated value is required to prevent any looping scenarios within the ring.
- Refer to the “[ERP Specifications](#)” on page 12-2 for timer defaults and valid ranges.

How Does ERP Work?

ERP operates over standard Ethernet interfaces that are physically connected in a ring topology. It uses an Automatic Protection Switching (APS) protocol to coordinate protection and recovery switching mechanisms over the Ethernet ring.

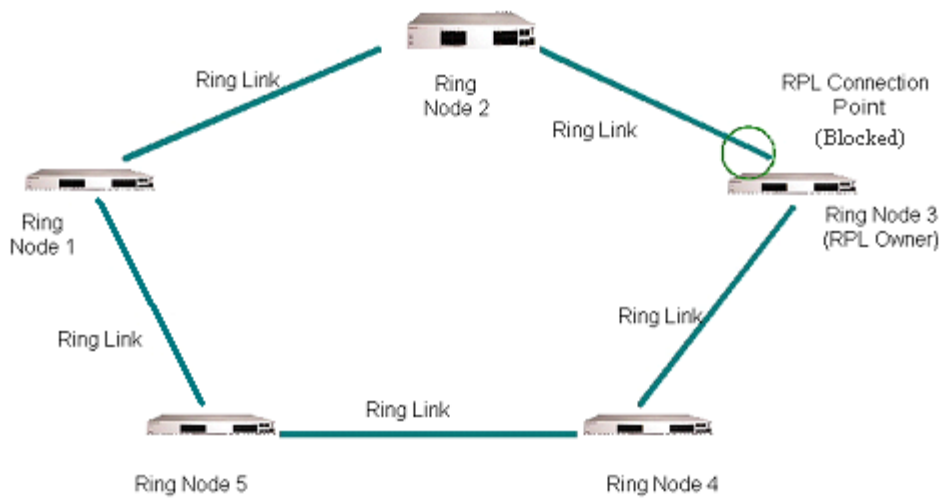
In an Ethernet ring, each node is connected to two adjacent nodes using two independent links called ring links. A ring link is bound by two adjacent nodes on ports called ring ports. The ring nodes support standard FDB (Filtering database) MAC learning, forwarding, flush behavior, and port blocking and unblocking mechanisms.

The Ethernet ring has a designated Ring Protection Link (RPL), which is blocked under normal conditions in order to avoid forming a loop in the ring. When a link or port failure is detected, a Signal Failure (SF) message is sent on the ring to inform other ring nodes of the failure condition. At this point the ring is operating in protection mode. When this mode is invoked, the RPL is unblocked forming a new traffic pattern on the ring, (for example, traffic is accommodated on the RPL but blocked on the failed link). The node responsible for blocking and unblocking the RPL is called the RPL Owner.

ERP Ring Modes

A ring operates in one of two modes: idle (normal operation; all links up and RPL is blocked) and protection (protection switching activated; a ring failure has triggered the RPL into a forwarding state).

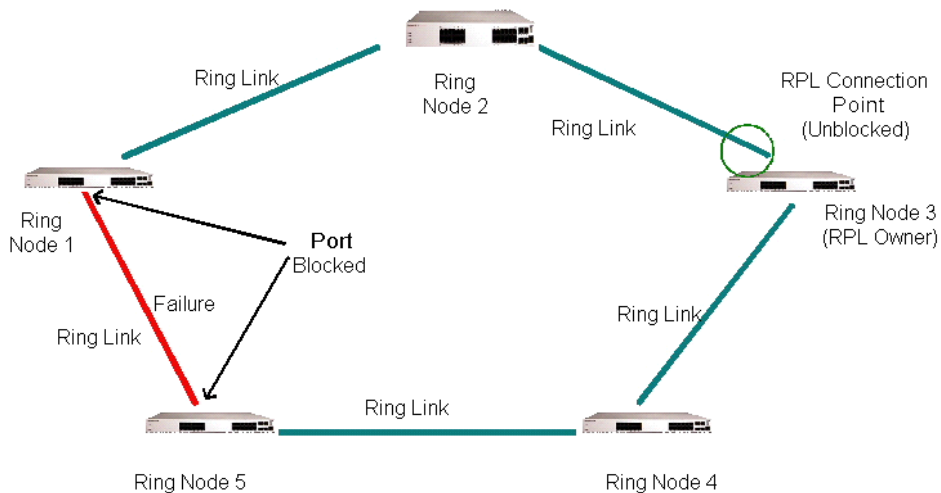
The following illustration shows an example of an ERP ring operating in the idle mode; all ring nodes are up and the RPL is blocked:



Normal Mode

If a link or node failure occurs in the ring shown in the above illustration, the ring transitions as follows into the protection mode:

- Nodes adjacent to the failure detect and report the failure using the R-APS (SF) message.
- The R-APS (SF) message triggers the RPL owner to unblock the RPL.
- All nodes in the ring flush all the dynamic MAC addresses learned on their ring ports.
- The ring is now operating in the protection mode, as shown below:



Protection Mode

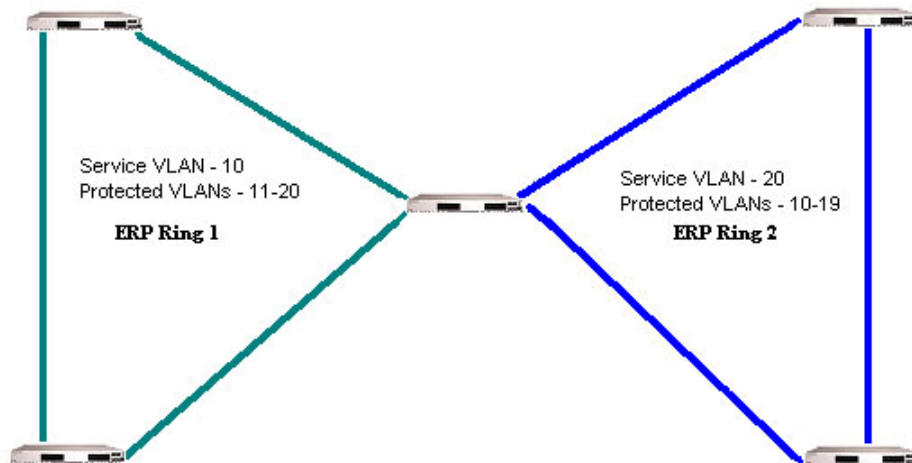
When the failed link shown in the above illustration recovers, the ring transitions as follows back to the idle mode:

- Nodes adjacent to the recovered link initiate an R-APS (NR) message and start the Guard Timer.

- When the RPL owner receives the R-APS (NR) message, it starts the Wait-To-Restore timer (WTR), which is the set period of time that must elapse before the RPL owner blocks the RPL.
- Once the WTR timer expires, the RPL owner blocks the RPL and transmits an R-APS (NR, RB) message indicating that RPL is blocked (RB).
- On receiving the R-APS (NR, RB) message, ring nodes flush all the dynamic MAC addresses learned on their ring ports and unblock any previously blocked ports.
- The ring is now operating in the idle mode. The RPL is blocked and all other ring links are operational.

Overlapping Protected VLANs Between ERP Rings on same Node

In a network where all connected nodes cannot belong to a single ERP ring, the OmniSwitch supports multiple ERP rings with a single shared node. The network example below shows two ERP rings connected with a shared node.



Overlapping Protected VLANs

In this example each of the ERP rings has a different Service VLAN configured which allows the ERP PDUs to be processed by the corresponding ERP ring node. The protected VLANs can be shared across ERP rings. The Service VLAN configured for ERP ring 1 is configured as a protected VLAN for ERP ring 2 and the Service VLAN for ERP ring 2 is configured as a protected VLAN for ERP ring 1.

- The two ERP rings must be configured with the same VLAN type; standard or VLAN stacking.
- Only a single node can be shared between the ERP rings; multiple shared nodes could result in a loop if the rings that are configured on shared nodes have common protected VLANs.
- Traffic for a protected VLAN will not be passed if the protected VLAN is deleted from either ERP ring or if the shared node goes down.

ERP and RRSTP Differences

ERP and the Ring Rapid Spanning Tree Protocol (RRSTP) are both used for the prevention of loops in ring-based topologies but have the following differences in their implementation and functionality:

- RRSTP uses a different destination MAC address for each ring, based on the ring ID. ERP uses the same destination MAC address for all ERP protocol frames and identifies the ring based on a unique Service VLAN associated with each ring, which carries the ERP protocol frames.
- When a link failure is detected, RRSTP quickly sets the blocking ports to a forwarding state but relies on MSTP for actual protocol convergence. ERP does not require any support from MSTP. ERP has an inherent mechanism to recover from a failed state once the failed link is active again.
- MSTP determines which ports of a fully active RRSTP ring are blocked. The blocked ports (Ring Protection Link) for an ERP ring is pre-determined and configured by the user.
- RRSTP requires a ring of contiguous RRSTP nodes. ERP allows non-ERP nodes to participate in the ring by using the connectivity monitoring capabilities of Ethernet OAM to alert ERP of a link failure through non-ERP nodes.

Interaction With Other Features

This section contains important information about ERP's interaction with other OmniSwitch features. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

Spanning Tree

ERP has the following interactions with Spanning Tree.

- Disabling Spanning Tree on the ring ports is required before changing the switch Spanning Tree operating mode from 1x1 to flat mode.

1X1 Mode

- 1X1 STP and ERP can co-exist on different ports on the same switch but not on the same VLAN-port association (VPA). STP continues to operate as usual on non-ERP ring ports even for the ERP Protected VLANs. On the ERP ring ports, the forwarding state is controlled by ERP.
- Maximum number of Protected VLANs supported is 252.

Flat Mode

- MSTP and ERP can co-exist on the same switch but are not supported on the same MSTI. ERP Protected VLANs can not be part of the same MSTI as non-ERP Protected VLANs.
- RSTP and ERP can co-exist on a node only if STP is disabled on ERP ports, the default-VLAN of ERP ports is disabled, and ERP protected VLANs are not configured on non-ERP ports. Also, non-ERP Protected VLANs should not be configured on ERP ports.
- RRSTP and ERP cannot be configured on the same port.

VLAN Stacking

The VLAN Stacking application has the following interactions with ERP:

- ERP is supported on Network Network Interface (NNI) ports; it is not supported on UNI ports.
- Tunneling of STP BPDUs across ERP links is not supported. However, tunneling of STP BPDUs across UNI ports is supported in a VLAN stacking configuration.

See [“Configuring ERP with VLAN Stacking NNIs” on page 12-16](#) for more information.

Ethernet OAM

ERP ring ports can be configured to accept a loss of connectivity event for a Remote Ethernet OAM Maintenance End Point (MEP). See [“Monitoring Remote Ethernet OAM End Points with ERP” on page 12-15](#) for more information.

Quick Steps for Configuring ERP with Standard VLANs

The following steps provide a quick tutorial for configuring ERP.

- 1 Create a VLAN using the **vlan** command.

```
-> vlan 1001
```

- 2 Create ERP ring ID 1, ERP Service VLAN and MEG Level and associate two ports to the ring using the **erp-ring** command.

```
-> erp-ring 1 port 1/1 port2 1/2 service-vlan 1001 level 5
```

- 3 Configure the RPL on one node using the **erp-ring rpl-node** command.

```
-> erp-ring 1 rpl-node port 1/1
```

- 4 Configure the protected VLANs using the **erp-ring protected-vlan** command.

```
-> erp-ring 1 protected-vlan 1002  
-> erp-ring 1 protected-vlan 1003-1005
```

- 5 Enable the ERP ring configuration using the **erp-ring enable** command.

```
-> erp-ring 1 enable
```

- 6 Display the ERP configuration using the **show erp** command.

```
-> show erp
```

Quick Steps for Configuring ERP with VLAN Stacking

The following steps provide a quick tutorial for configuring ERP with VLAN Stacking:

- 1 Create a VLAN Stacking SVLAN 1001 using the **ethernet-service** command.

```
-> ethernet-service svlan 1001
```

- 2 Create a VLAN Stacking service and associate the service with SVLAN 1001 using the **ethernet-service service-name** command.

```
-> ethernet-service service-name CustomerA svlan 1001
```

- 3 Configure ports 1/1 and 1/2 as VLAN Stacking Network Network Interface (NNI) ports, associate the ports with SVLAN 1001, and configure them for use with ERP using the **ethernet-service svlan nni** command.

```
-> ethernet-service svlan 1001 nni 1/1 erp
-> ethernet-service svlan 1001 nni 1/2 erp
```

- 4 Create ERP ring ID 1 and associate the two NNI ports to the ring using the **erp-ring** command.

```
-> erp-ring 1 port 1/1 port2 1/2 service-vlan 1001 level 5
```

- 5 Configure the RPL on one node using the **erp-ring rpl-node** command.

```
-> erp-ring 1 rpl-node port 1/1
```

- 6 Create additional SVLANs to add to the ring using the **ethernet-service** command.

```
-> ethernet-service svlan 1002
-> ethernet-service svlan 1003
-> ethernet-service svlan 1004
-> ethernet-service svlan 1005
```

- 7 Configure the SVLANs created in Step 6 as ERP protected VLANs using the **erp-ring protected-vlan** command.

```
-> erp-ring 1 protected-vlan 1002-1005
```

Note that when two VLAN Stacking NNI ports are associated with the same SVLAN and both those ports will serve as the ring ports for the node, the SVLAN is automatically added to the list of protected SVLANs for the ERP ring. For example, the following commands designate SVLAN 1002 as a protected VLAN:

```
-> ethernet-service svlan 1002 nni 1/1 erp
-> ethernet-service svlan 1002 nni 1/2 erp
```

- 8 Enable the ERP ring configuration using the **erp-ring enable** command.

```
-> erp-ring 1 enable
```

- 9 Display the ERP configuration using the **show erp** command.

```
-> show erp
```


ERP Configuration Overview and Guidelines

Configuring ERP requires several steps. These steps are outlined here and further described throughout this section. For a brief tutorial on configuring ERP, see [“Quick Steps for Configuring ERP with Standard VLANs” on page 12-9](#).

By default, ERP is disabled on a switch. Configuring ERP consists of these main tasks:

- 1** Configure the basic components of an ERP ring (ring ports, service VLAN, and MEG level). See [“Configuring an ERP Ring” on page 12-12](#).
- 2** Tag VLANs for ring protection. See [“Adding Protected VLANs” on page 12-13](#).
- 3** Configure an RPL port. When a ring port is configured as an RPL port, the node to which the port belongs becomes the RPL owner. The RPL owner is responsible for blocking and unblocking the RPL. See [“Configuring an RPL Port” on page 12-13](#).
- 4** Change the Wait-To-Restore timer value. This timer value determines how long the RPL owner waits before restoring the RPL to a forwarding state. See [“Setting the Wait-to-Restore Timer” on page 12-14](#).
- 5** Change the Guard timer value. This timer value determines an amount of time during which ring nodes ignore R-APS messages. See [“Setting the Guard Timer” on page 12-14](#).
- 6** Configure the ring port to receive the loss of connectivity event for a Remote Ethernet OAM endpoint. See [“Monitoring Remote Ethernet OAM End Points with ERP” on page 12-15](#).
- 7** Configure a VLAN Stacking NNI-to-SVLAN association for ERP control. This is done to include an SVLAN in a ring configuration. See [“Configuring ERP with VLAN Stacking NNIs” on page 12-16](#).
- 8** Clear ERP statistics. Commands to clear ERP statistics for a single ring or multiple rings are described in [“Clearing ERP Statistics” on page 12-18](#).

Configuration Guidelines

Use the following guidelines when configuring ERP for the switch:

- Physical switch ports and logical link aggregate ports can be configured as ERP ring ports. This also includes VLAN Stacking Network Network Interface (NNI) ports.
- ERP is *not* supported on mobile ports, mirroring ports, link aggregate member ports, high availability ports, multicast VLAN receiver ports (ERP is supported on Multicast VLAN sender ports only), VLAN Stacking User Network Interface (UNI) ports, or RRSTP ring ports.
- An ERP ring port can belong to only one ERP ring at a time.
- When configuring a ring for a switch that is operating in the flat Spanning Tree mode using STP or RSTP (not MSTP), administratively disable the default VLAN for the ring port. In this case, if the switch is using RSTP, disabling Spanning Tree on the ring port is also required.
- When configuring a ring for a switch that is operating in the flat Spanning Tree mode using MSTP, make sure the standard VLAN to which the ring port is assigned is not a member of an MSTI that is also associated with ERP protected VLANs.
- The specified service VLAN ID must not participate in a Spanning Tree instance that is associated with non-ERP VLANs. This may require changing the Spanning Tree configuration for the VLAN ID prior to using this command.

- If the ERP switch participates in an Ethernet OAM MaintenanceDomain(MD), configure the Management Entity Group (MEG) level of the ERP service VLAN with the number that is used for the Ethernet OAM MD.
- The Service VLAN can belong to only one ERP ring at a time and must be a static VLAN. Note that the service VLAN is also a protected VLAN.

Configuring an ERP Ring

The following configuration steps are required to create an ERP ring:

- 1** Determine which two ports on the switch will become the ring ports. For example, ports 1/2 and 3/1.
- 2** Administratively disable the VLAN that will serve as the default VLAN for the ring ports if the switch is operating in the flat Spanning Tree mode without MSTP. For example, if VLAN 10 is the default VLAN for ports 1/2 and 3/1, before configuring 1/2 and 3/1 as a ring ports, disable VLAN 10.

```
-> vlan 10 disable
```

- 3** Disable Spanning Tree on the ports that will become the ring ports if the switch is operating in the flat Spanning Tree mode and using RSTP. For example, disable the Spanning Tree for the VLAN 10 port 1/2 instance and VLAN 10 port 3/1 instance:

```
-> bridge 10 1/2 disable  
-> bridge 10 3/1 disable
```

- 4** Determine which VLAN on the switch will become the ERP service VLAN for the ring. If the VLAN does not exist, create the VLAN. For example:

```
-> vlan 500
```

- 5** Determine the APS Management Entity Group (MEG) level number to assign to the service VLAN. If the ERP switch participates in an Ethernet OAM MaintenanceDomain(MD), configure the MEG level with the same number used for the Ethernet OAM MD.

- 6** Create the ERP ring configuration on each switch using the **erp-ring** command. For example the following command configures an ERP ring with ring ID 1 on ports 1/2 and 3/1 along with service VLAN 1001 and MEG level 2.

```
-> erp-ring 1 port1 1/2 port2 3/1 service-vlan 500 level 2  
-> erp-ring 1 enable
```

To configure link aggregate logical ports as ring ports, use the **erp-ring** command with the **linkagg** parameter. For example:

```
-> erp-ring 1 port1 linkagg 1 port2 linkagg 2 service-vlan 1001 level 2  
-> erp-ring 1 enable
```

- 7** Repeat Steps 1 through 6 for each switch that will participate in the ERP ring. Make sure to use the same VLAN ID and MEG level for the service VLAN on each switch.

Use the **show erp** command to verify the ERP ring configuration. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Removing an ERP Ring

To delete an ERP ring from the switch configuration, use the **no** form of the **erp-ring** command. For example:

```
-> no erp-ring 1
```

Note. Administratively disable ring ports before deleting the ring to avoid creating any network loops. Once a ring is deleted, then administratively enable the ports under Spanning Tree protocol.

Adding Protected VLANs

ERP allows a single VLAN or a number of VLANs to participate in a single ERP ring. The **erp-ring protected-vlan** command is used to tag the ring ports of the ERP ring with a VLAN ID. Once a VLAN is associated with a ring, it is referred to as an ERP protected VLAN.

An ERP ring must already exist before protected VLANs are added to the ring. Similarly, the VLAN must already exist before it is added as a protected VLAN to the ring.

To configure a VLAN or range of VLANs as protected VLANs for a specific ring, enter **erp-ring** followed by a ring ID then **protected-vlan** followed by a single VLAN ID or a range of VLAN IDs. **For example:**

```
-> erp-ring 1 protected-vlan 11
-> erp-ring 1 protected-vlan 12-20 25-40 100
```

To delete a protected VLAN or group of VLANs from the ring, use the **no** form of the **erp-ring protected-vlan** command. For example:

```
-> no erp-ring 1 protected-vlan 11
-> no erp-ring 1 protected-vlan 31-40
```

Use the **show erp protected-vlan** command to view the protected VLANs. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Configuring an RPL Port

A ring protection link (RPL) port can be a physical or logical port. The port must be a ring port before it is configured as an RPL port, and out of the two ring ports on the node, only one can be configured as a RPL port. The RPL remains blocked to prevent loops within the ERP ring.

To configure an RPL port, first disable the ring and then use the **erp-ring rpl-node** command to specify which ring port will serve as the RPL. For example:

```
-> erp-ring 1 disable
-> erp-ring 1 rpl-node port 1/1
-> erp-ring 1 enable
```

Note. RPL node can be configured only when the ring is disabled; RPL configuration applied to the ring while it is enabled will be rejected.

To remove the RPL node configuration for the specified ring, use the **no** form of the **erp-ring rpl-node** command. For example:

```
-> no erp-ring 1 rpl-node
```

To verify the RPL node configuration for the switch, use the **show erp** command. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Setting the Wait-to-Restore Timer

The wait-to-restore (WTR) timer determines the number of minutes the RPL owner waits before blocking the RPL port after the ERP ring has recovered from a link failure.

By default, the WTR time is set to five minutes. To change the value of the WTR timer, use the **erp-ring wait-to-restore** command. For example:

```
-> erp-ring 1 wait-to-restore 6
```

The above command is only used on a switch that serves as the RPL node for the ERP ring. The specified ERP ring ID must already exist in the switch configuration.

To restore the timer back to the default setting, use the **no** form of the **erp-ring wait-to-restore** command. For example:

```
-> no erp-ring 1 wait-to-restore
```

To verify the WTR configuration, use the **show erp** command. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Setting the Guard Timer

The guard timer is used to prevent the ring nodes from receiving outdated R-APS messages, which are no longer relevant. Receiving outdated R-APS messages could result in incorrect switching decisions. During the amount of time determined by this timer, all received R-APS messages are ignored by the ring protection control process.

By default, the guard timer value is set to 50 centi-seconds. To change the value of this timer, use the **erp-ring guard-timer** command. For example:

```
-> erp-ring 1 guard-timer 100
```

To restore the Guard Timer back to the default value, use the **no** form of the **erp-ring guard-timer** command. For example:

```
-> no erp-ring 1 guard-timer
```

To verify the configured Guard Timer, use the **show erp** command. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Monitoring Remote Ethernet OAM End Points with ERP

By default, ERP ring ports drop loss of connectivity events for a Remote Ethernet OAM Maintenance End Point (MEP). Configuring the ring port to accept such events allows ERP to interact with Ethernet OAM and monitor non-ERP nodes that may exist in the ring.

The **erp-ring ethoam-event remote-endpoint** command is used to configure a ring port to accept or deny loss of connectivity events. Note that following conditions are required before this command is allowed:

- An Ethernet OAM Maintenance Domain (MD) exists, and the ERP ring Maintenance Entity Group (MEG) level value is configured with the same number used for the MD level value.
- An Ethernet OAM Maintenance Association (MA) is present on the service VLAN for the ring.
- A down MEP is created on the port before the port is configured as a ring port.
- The Remote MEP-ID (RMEP-ID) is present in the MEP-LIST and the RMEP-ID specified is different from the down MEP ID configured for the ring port.

For more information about configuring the Ethernet OAM components mentioned above, see [Chapter 15, “Configuring Ethernet OAM.”](#)

To configure a ring port to accept loss of connectivity events, enter **erp-ring** followed by an existing ring ID number, **ethoam-event port** followed by the ring port number, then **remote-endpoint** followed by the remote MEP ID number. For example:

```
-> erp-ring 1 ethoam-event port 1/1 remote-endpoint 10
```

The above command configures ring port 1/1 on ERP ring 1 to accept loss of connectivity events from remote endpoint 10.

The **erp-ring ethoam-event remote-endpoint** command is also used to configure a link aggregate logical port to accept or drop loss of connectivity events. For example:

```
-> erp-ring 1 ethoam-event linkagg 1 remote-endpoint 20
```

To configure the ERP ring port to drop loss of connectivity events, use the **no** form of the **erp-ring ethoam-event remote-endpoint** command. For example:

```
-> no erp-ring 1 ethoam-event port 1/1
```

To verify the Ethernet OAM event configuration for a specific ring port, use the **show erp** command with the **port** parameter. For example:

```
-> show erp port 1/15
Legend: * - Inactive Configuration

Ring-Id : 1
  Ring Port Status      : forwarding,
  Ring Port Type        : non-rpl,
  Ethoam Event          : disabled
```

For more information about these commands, see the *OmniSwitch CLI Reference Guide*.

Configuring ERP with VLAN Stacking NNIs

A VLAN Stacking Network Network Interface (NNI) can participate in an ERP ring. However, an NNI is created through an association of a port with an SVLAN. Both STP and ERP cannot control the same VLAN-port association (VPA). By default, the NNI to SVLAN association is controlled by STP.

To include an NNI in an ERP ring, specify ERP control at the time the NNI association is configured. This is done using the **erp** parameter of the **ethernet-service svlan nni** command. For example:

```
-> ethernet-service svlan 1001 nni 1/1 erp
-> ethernet-service svlan 1001 nni 1/2 erp
```

The above commands configure ports 1/1 and 1/2 as NNI ports for SVLAN 1001 with ERP control over the VPA. Note that the SVLAN specified must already exist in the switch configuration.

To configure an ERP ring with NNI-SVLAN associations, use the **erp-ring** command but specify an SVLAN ID for the service VLAN and the associated NNI ports as the ring ports. For example:

```
-> erp-ring 1 port1 1/1 port2 1/2 service-vlan 1001 level 2
-> erp-ring 1 enable
```

Note the following when configuring an ERP ring with VLAN Stacking NNI-SVLAN associations:

- Only two ERP type NNI associations are allowed per SVLAN.
- Configuring an ERP ring on 8021q tagged port associations with SVLANs is not allowed.
- Configuring an ERP Ring on an STP type NNI association with an SVLAN is not allowed.
- Configuring an IMPVLAN as an ERP service VLAN is not allowed.
- If an SVLAN that is not associated with any NNI ports is configured as the service VLAN for an ERP ring, the NNI ring ports are automatically associated with that SVLAN at the time the ring is created.
- SVLAN User Network Interface (UNI) associations are not eligible for ERP ring protection.
- If the ERP type NNI ports are connected to the STP path via UNI ports, then STP BPDUs can be tunneled with the help of VLAN-stacking mechanism.
- Deleting an ERP service VLAN and its associated NNI ports is only allowed when the ERP ring itself is deleted using the **no** for of the **erp-ring** command. None of the VLAN Stacking CLI commands can remove a service VLAN consisting of an NNI-SVLAN association.

The following sequence of configuration commands provides an example of configuring an ERP ring consisting of VLAN Stacking NNI ports and SVLANs:

```
-> ethernet-service svlan 100
-> ethernet-service svlan 200
-> ethernet-service svlan 100 nni 1/3
-> ethernet-service svlan 100 nni 1/1 erp
-> ethernet-service svlan 100 nni 1/2 erp
-> erp-ring 10 port1 1/1 port2 1/2 service-vlan 200 level 3 enable
```

In the above example, ERP ring 10 is configured as follows:

- 1** SVLANs 100 and 200 are created.
- 2** Port 1/3 is associated with SVLAN 100, but no **erp** parameter is used. As a result, port 1/3 is an STP type NNI association by default.

- 3 Ports 1/1 and 1/2 are associated with VLAN 100 using the **erp** parameter. These ports are now ERP type NNI associations.
- 4 The ERP ring is created specifying NNI ports 1/1 and 1/2 as the ring ports, SVLAN 200 as the service VLAN, and an MEG level of 3.
- 5 When ERP ring 10 is created, ERP type NNI associations are automatically configured between the ring ports and SVLAN 200. Note that prior to creating this ring, SVLAN 200 had no configured NNI associations.

Configuring ERP Protected SVLANs

An SVLAN becomes an ERP protected SVLAN when any one of the following actions occur:

- The **erp-ring protected-vlan** command is used to explicitly add an SVLAN to the ring.
- The **ethernet-service vlan nni** command is used to configure an ERP type SVLAN association with two NNI ports that also serve as ring ports. In this case, the SVLAN is automatically protected as part of the association with NNI ring ports.

The following sequence of configuration commands provides an example of how SVLANs are automatically added as protected SVLANs to an ERP ring:

```
-> ethernet-service vlan 100
-> ethernet-service vlan 200
-> ethernet-service vlan 300
-> ethernet-service vlan 400
-> ethernet-service vlan 100 nni 1/1 erp
-> ethernet-service vlan 100 nni 1/2 erp
-> ethernet-service vlan 200 nni 1/1 erp
-> ethernet-service vlan 200 nni 1/2 erp
-> ethernet-service vlan 300 nni 1/1 erp
-> ethernet-service vlan 300 nni 1/2 erp
-> erp-ring 10 port1 1 1/1 port 2 1/2 service-vlan 400 level 2
```

In the above example:

- SVLANs 100 and 200 are automatically added as protected VLANs when the ring is created. This is due to the configuration of ERP type NNI associations between these SVLANs and ports 1/1 and 1/2, which become the ring ports for ERP ring 10.
- SVLAN 400 is also automatically added as a protected VLAN when it is configured as the service VLAN for the ring.
- SVLAN 300 is not added as a protected SVLAN because it is configured with an STP type NNI association.

As an alternative, the user could have manually added SVLANs 100 and 200 as protected SVLANs to ring 10 using the **erp-ring protected-vlan** command.

Use the **show erp** command to verify the configured VLAN Stacking ERP ring configuration. For more information about these commands, see the *OmniSwitch CLI Reference Guide*.

Clearing ERP Statistics

To clear ERP statistics for all rings in the switch, use the **clear erp statistics** command. For example:

```
-> clear erp statistics
```

To clear ERP statistics for a specific ring in the switch, use the **clear erp statistics** command with the **ring** parameter to specify a ring ID. For example:

```
-> clear erp statistics ring 5
```

To clear ERP statistics for a specific ring port, use the **clear erp statistics** command with the **ring** and **port** parameters. For example:

```
-> clear erp statistics ring 5 port 1/2
```

To clear ERP statistics for a specific link aggregate ring port, use **clear erp statistics** command with the **ring** and **linkagg** parameters. For example:

```
-> clear erp statistics ring 5 linkagg 2
```

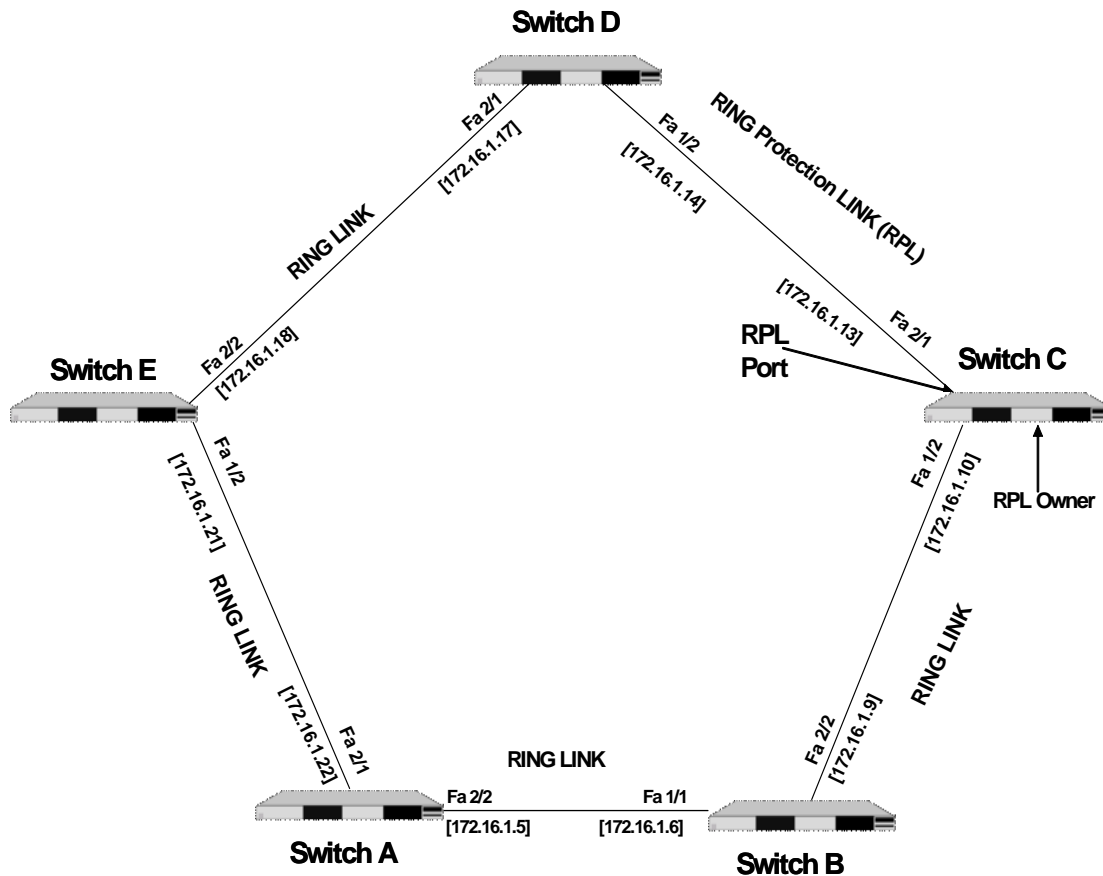
Use the **show erp statistics** command to verify ERP statistics. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Sample Ethernet Ring Protection Configuration

This section provides an example network configuration in which ERP is configured on network switches to maintain a loop-free topology. In addition, a tutorial is also included that provides steps on how to configure the example network topology using the Command Line Interface (CLI).

Example ERP Overview

The following diagram shows a five-switch ERP ring configuration:



Configuring the sample ERP ring network shown in the above diagram involves the following tasks:

- 1 Configure an ERP ring with ERP ring ID 1 on all switches in the network.
- 2 Define an ERP Service VLAN as VLAN 10 on all switches.
- 3 Set the Management Entity Group (MEG) level to 2 for all switches.
- 4 Switch C is the RPL owner; configure the port connected to the Ring Protection Link as a RPL port.
- 5 Enable the configured ERP ring.
- 6 Assign VLANs 11-20 as a protected VLANs to ERP ring 1.
- 7 Use the default settings for the guard timer and WTR timer values. These values can be adjusted as necessary.

Example ERP Configuration Steps

The following steps provide a quick tutorial for configuring the ERP ring network shown in the diagram on [page 12-19](#):

- 1 Configure ERP ring 1 and add protected VLANs 11 through 20 on Switch A, B, C, D, and E using the following commands:

```
-> erp-ring 1 port1 2/1 port2 2/2 service-vlan 10 level 2
-> erp-ring 1 enable
-> erp-ring 1 protected-vlan 11 - 20
```

- 2 Configure Switch C as the RPL owner for the ring using the following commands to designate port 2/1 as the RPL port:

```
-> erp-ring 1 disable
-> erp-ring 1 rpl-node port 2/1
-> erp-ring 1 enable
```

- 3 Verify the ERP ring configuration on any switch using the following command:

```
-> show erp ring 1
Legend: * - Inactive Configuration

Ring Id           : 1,
Ring Port1        : 2/1,
Ring Port2        : 1/2,
Ring Status       : enabled,
Service VLAN      : 10,
WTR Timer (min)   : 5,
Guard Timer (centi-sec) : 50,
MEG Level         : 2,
Ring State        : idle,
Ring Node Type    : rpl,
RPL Port          : 2/1,
Last State Change : SUN DEC 25 06:50:17 2016 (sysUpTime 00h:01m:31s)
```

The above output example shows that ERP ring 1 is created on ring ports 2/1 and 1/2 with service VLAN 10, WTR timer of 5 mins, Guard timer of 50 centi-seconds, MEG level 2, and port 2/1 is the RPL port.

- 4 Verify that VLANs 11 through 20 are protected VLANs for ERP ring 1 using the following command:

```
-> show erp ring 1 protected-vlan
Ring Id           : 1,
Protected VLAN    : 11-20,
```

- 5 Verify the status of an ERP ring port on any switch using the following command:

```
-> show erp port 1/2
Legend: * - Inactive Configuration

Ring-Id : 1
Ring Port Status : forwarding,
Ring Port Type   : non-rpl,
Ethoam Event     : disabled
```

The above command shows the forwarding status of the port, the type of ring port (RPL or non-RPL), and ETHOAM event status.

Verifying the ERP Configuration

A summary of the **show** commands used for verifying the ERP configuration is given here:

show erp	Displays the ERP configuration information for all rings, a specific ring, or for a specific ring port.
show erp protected-vlan	Displays the protected VLAN configuration for all ERP rings or for a specific ring.
show erp statistics	Displays the ERP statistics for all rings, a specific ring, or a specific ring port.
show ethernet-service	Displays configuration information for VLAN Stacking Ethernet services, which includes SVLANs and NNI port associations.
show ethernet-service nni	Displays the VLAN Stacking NNI configuration.
show ethernet-service vlan	Displays a list of SVLANs configured from the switch.

For more information about the displays that result from these commands, see the *OmniSwitch CLI Reference Guide*.

13 Configuring Loopback Detection

Loopback Detection (LBD) automatically detects and prevents forwarding loops on ports and Link Aggregations (LAGs) that have forwarded network traffic which has looped back to the originating switch. LBD detects and prevents Layer 2 forwarding loops on a port either in the absence of other loop detection mechanisms such as STP/RSTP/MSTP, or when these mechanisms cannot detect it (e.g., a client's equipment may drop BPDUs, or the STP protocol may be restricted to the network edge).

In This Chapter

This chapter describes the LBD feature and how to configure it through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*. This chapter provides an overview of LBD and includes the following information:

- [“LBD Specifications” on page 13-1](#)
- [“Quick Steps for Configuring LBD” on page 13-3](#)
- [“LBD Overview” on page 13-4](#)
- [“Configuring LBD” on page 13-5](#)
- [“Verifying the LBD Configuration” on page 13-7](#)

LBD Specifications

RFCs supported	Not applicable at this time
IEEE Standards Supported	Not applicable at this time
Platforms Supported	OmniSwitch 6250-Metro Models
Ports Supported	There is no restriction on type of ports on which the LBD can be enabled. But it is recommended LBD should be enabled on the edge ports.
Transmission Timer	The valid range is from 5 to 600 seconds.
Autorecovery Timer	The valid range is from 30 to 86400 seconds.

LBD Defaults

The following table shows LBD default values.

Parameter Description	Command	Default Value/Comments
LBD administrative state	loopback-detection	Disabled
LBD status of a port	loopback-detection port	Disabled
Transmission time is the time period between LBD packet transmissions.	loopback-detection transmission-timer	30 seconds
Autorecovery timer	loopback-detection autorecovery-timer	300 seconds

Quick Steps for Configuring LBD

The following steps provide a quick tutorial on how to configure LBD. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

- 1 To enable the LBD protocol on a switch, use the **loopback-detection** command. For example:

```
-> loopback-detection enable
```

- 2 To enable the LBD protocol on a port, use the **loopback-detection port** command by entering **LBD port**, followed by the slot and port number, and enable. For example:

```
-> loopback-detection port 1/1 enable
```

- 3 Configure the LBD transmission timer by using the **loopback-detection transmission-timer** command. For example:

```
-> loopback-detection transmission-timer 200
```

- 4 Configure the LBD autorecovery timer by using the **loopback-detection autorecovery-timer** command. For example:

```
-> loopback-detection autorecovery-timer 300
```

Note. *Optional.* Verify the LBD global configuration by entering the **show loopback-detection** configuration command or verify the LBD configuration on a port by entering the **show loopback-detection port** command. For example:

```
-> show loopback-detection
```

```
Global LBD Status           : Enabled
Global LBD Transmission Timer : 200 sec
Global LBD Auto-recovery Timer : 300 sec
```

```
-> show loopback-detection port 1/1
```

```
Global LBD Status           : Enabled
Global LBD Transmission Timer : 200 sec
Global LBD Autorecovery Timer : 300 sec
Port LBD Status             : Enabled
Port LBD State               : Normal
```

To verify the LBD statistics of a port, use the **show loopback-detection statistics port** command. For example:

```
-> show loopback-detection statistics port 1/1
```

```
LBD Port Statistics
LBD Packet Send           : 1
Invalid LBD Packet Received : 0
```

LBD Overview

Loopback Detection (LBD) automatically detects and prevents L2 forwarding loops on a port. LBD operates in addition to STP which detects forwarding loops. When a loopback is detected, the port is disabled and goes into a shutdown state. Once a loop is discovered, the port from where the loops originated is placed into shutdown state. A trap is sent and the event is logged. Network managers can define a Recovery Interval which automatically places the port into a normal state, after the defined time period.

When enabling and configuring Loopback Detection:

- Enable Loopback Detection globally on the switch.
- Enable Loopback Detection on edge port.

The switch periodically sends out LBD frame from loopback detection enabled port and concludes that the port is looped back if it receives the frame on any of the loop-back detection enabled ports.

Transmission Timer

Transmission timer is the time duration in seconds at which the port sends LBD frame on the link. When any of the port is getting blocked due to loopback detection, there will be no further transmission and receiving of any traffic on the blocked port. The port will be go to shutdown state.

Autorecovery

When the ports are shutdown due to LBD, the auto recovery mechanism moves the ports to a normal state after a specific time period. Autorecovery is available on all the ports that have been disabled due to loopback detection and also be configured on the switch by using CLI command. The autorecovery time period can be configured globally on the switch.

Interaction With Other Features

This section contains important information about how other OmniSwitch features interact with LBD. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

Spanning Tree Protocol

- If the STP mode is set to Multiple Spanning Tree, Loopback Detection can only be enabled on interfaces where STP is disabled.
- LBD frame are always sent untagged regardless of the spanning tree state on the port.

Link Aggregation

When loopback is detected on any one of the Linkagg port, all the ports of the linkagg will be shutdown due to loopback detection.

Configuring LBD

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to configure LBD on a switch.

- Enable LBD on a switch or port (see [“Enabling LBD” on page 13-5](#))
- Configure the LBD transmission timer (see [“Configuring the LBD Transmission Timer” on page 13-5](#))
- Configure the autorecovery timer (see [“Configuring the Autorecovery Timer” on page 13-5](#))
- View the LBD statistics on a port (see [“Viewing LBD Statistics” on page 13-5](#))
- Recover a port from LBD shutdown (see [“Recovering a Port from LBD Shutdown” on page 13-6](#))

Enabling LBD

By default, LBD is disabled on the switch. To enable LBD on a switch, use the **loopback-detection** command. For example, the following command enables LBD on a switch:

```
-> loopback-detection enable
```

Enabling LBD on a Port

By default, LBD is disabled on all switch ports. To enable LBD on a port, use the **loopback-detection port** command. For example, the following command enables LBD on port 1 of slot 1:

```
-> loopback-detection port 1/1 enable
```

To enable LBD on multiple ports, specify a range of ports. For example:

```
-> loopback-detection port 1/1-8 enable
```

Configuring the LBD Transmission Timer

To configure the transmission time period between LBD packet transmissions, use the **loopback-detection transmission-timer** command. For example:

```
-> loopback-detection transmission-timer 200
```

Configuring the Autorecovery Timer

To configure the LBD autorecovery timer on the switch, use the **loopback-detection autorecovery-timer** command. For example:

```
-> loopback-detection autorecovery-timer 300
```

Viewing LBD Statistics

To view the LBD statistics on a specific port, use the **show loopback-detection statistics port** command. For example, to view the statistics for port 1 on slot 1, enter:

```
-> show loopback-detection statistics port 1/1
```

Recovering a Port from LBD Shutdown

To bring a port out of the shutdown state, use the **interfaces clear-violation-all** command. For example, to bring port 5 on slot 1 out of the shutdown state, enter:

```
-> interfaces 1/5 clear-violation-all
```

To bring multiple ports out of the shutdown state, enter:

```
-> interfaces 5/5-10 clear-violation-all
```

Verifying the LBD Configuration

To display LBD configuration and statistics information, use the show commands listed below:

show loopback-detection	Displays the global LBD configuration information for the switch.
show loopback-detection port	Displays LBD configuration information for all ports on the switch.
show loopback-detection statistics port	Displays LBD statistics information for a specific port on the switch..

Note. For more information about the resulting display from these commands, see the *OmniSwitch CLI Reference Guide*.

14 Configuring CPE Test Head

The Customer Provider Edge (CPE) Test Head traffic generator and analyzer is a Test-OAM (Operation, Administration and Maintenance) tool used in the Metro Ethernet Network to validate the customer Service Level Agreements (SLA). This functionality allows the operator to validate the Metro Ethernet Network between customer end points, which is critical when provisioning or troubleshooting network services.

In This Chapter

This chapter describes the CPE Test Head feature and how to configure it through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*. This includes the following information:

- [“CPE Test Head Specifications” on page 14-2](#)
- [“Quick Steps for Configuring CPE Test Head” on page 14-3](#)
- [“CPE Test Head Overview” on page 14-5](#)
- [“CPE Test Head Configuration Overview” on page 14-6](#)
- [“Configuring a CPE Test Profile” on page 14-7](#)
- [“Running a CPE Test” on page 14-9](#)
- [“Verifying the CPE Test Configuration and Results” on page 14-10](#)

CPE Test Head Specifications

Platforms Supported	OmniSwitch 6250-Metro Models
Tests supported	Unidirectional throughput test
Maximum number of tests per switch	32
Number of active tests allowed per switch	1
Supported test roles	Generator or Analyzer (Only one role per test; switch cannot perform both roles for the same test)
Test mode supported	ingress UNI
Test traffic direction supported	unidirectional

Quick Steps for Configuring CPE Test Head

The following steps provide a quick tutorial on how to configure a CPE test profile and run a CPE test. Each step describes a specific operation and provides the CLI command syntax that is used to perform that operation.

Configure the Test Profile

The CPE test profile is configured on both the generator and analyzer switch. Steps 1 through 4 configure profile parameters common to both the generator and analyzer switch. Steps 5 through 7 configure profile parameters required only for the generator.

- 1 Configure the name for the CPE test, use the **test-oam** command. For example:

```
-> test-oam Test1 descr First-test
```

- 2 Configure the source and destination end point for the test, use the **test-oam src-endpoint dst-endpoint** command. For example:

```
-> test-oam Test1 src-endpoint SW1
```

```
-> test-oam Test1 dst-endpoint SW2
```

- 3 Configure the source MAC address, destination MAC address and the SVLAN for the test frame using the **test-oam vlan test-frame** command. For example:

```
-> test-oam Test1 vlan 100 test-frame src-mac 00:00:00:00:00:01 dst-mac  
00:00:00:00:00:02
```

- 4 Configure the type of role the switch will perform using the **test-oam role** command. For example:

```
-> test-oam Test1 role generator
```

- 5 Configure the test port on the generator switch using the **test-oam port** command. For example:

```
-> test-oam Test1 port 1/1
```

- 6 Configure the test packet parameters using the **test-oam frame** command. For example:

To configure a Layer 2 test frame, specify a hexadecimal Ether type value.

```
-> test-oam Test1 frame vlan-tag 1 priority 2 drop-eligible false ether-type  
0x0100 data-pattern 0x0010
```

To configure a Layer 3 test frame, specify **ipv4** as the Ether type value.

```
-> test-oam Test1 frame vlan-tag 1 priority 2 drop-eligible false ether-type  
ipv4 src-ip 1.1.1.1 dst-ip 2.2.2.2 ttl 4 tos 0x01 protocol udp src-port 2000  
dst-port 3000 data-pattern 0x0010
```

- 7 Configure the test duration, rate and packet-size using the **test-oam duration rate packet-size** command. For example:

```
-> test-oam Test1 duration 10 rate 8kbps packet-size 64
```

Running the Test

1 Start the test on the analyzer switch first and then on the generator switch using the **start** option of the **test-oam start stop** command. For example:

```
-> test-oam Test1 start
```

When the test runs the amount of time specified for the test duration, the test automatically stops.

2 To stop an active test from running, use the **stop** form of the **test-oam start stop** command. For example:

```
-> test-oam Test1 start
```

Note. Verify the test configuration and status with the **show test-oam** command. For example:

```
-> show test-oam tests
Total Test-Ids: 1
Test-Id      Port  Src-Mac          Dst-Mac          Vlan  Direction      Status
-----+-----+-----+-----+-----+-----+-----
Test1       1/1  00:00:00:00:00:01 00:00:00:00:00:02  100  Unidirectional  ended
```

To verify test results, use the **show test-oam statistics** command. For example:

```
-> show test-oam Test1 statistics
Test-Id      TX-Ingress      TX-Egress      RX-Ingress
-----+-----+-----+-----
Test1              1200366        1200366          0
```

To clear test statistics, use the **clear test-oam statistics** command. For example:

```
-> clear test-oam Test1 statistics
This clears all the statistics related to "Test1".
```

```
-> clear test-oam statistics
This will clear the statistics for all the tests.
```

See the *OmniSwitch CLI Reference Guide* for more information about these commands.

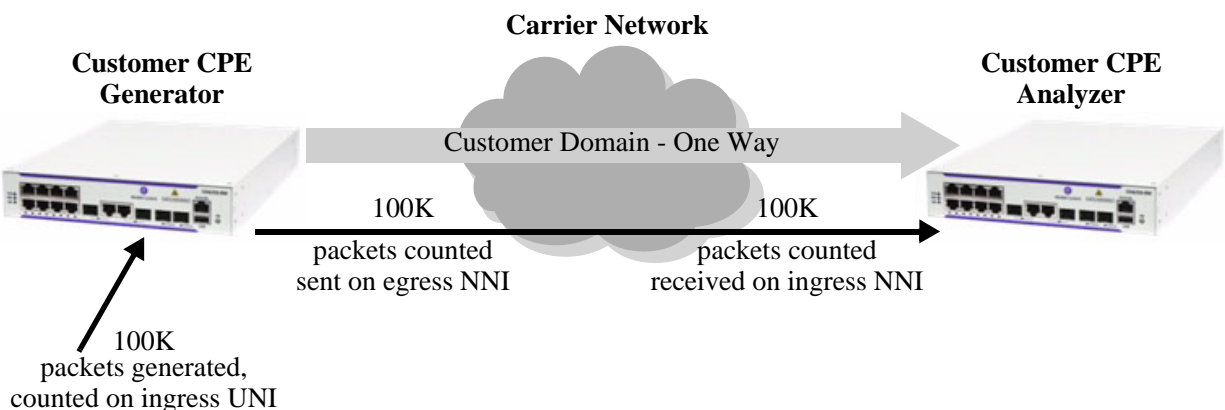
CPE Test Head Overview

The OmniSwitch CPE Test Head feature provides a remote test generator and analyzer capability for testing and validating the customer Ethernet service domain from end-to-end. This allows the service provider to perform the following tasks without the need for an external test head device:

- Generate specific flow-based traffic across the customer's Ethernet Virtual Circuit (EVC) to help identify flow-based issues.
- Identify the impact of QoS settings (SAP profile or QoS policies) on the overall traffic.
- Confirm throughput across the provider network.
- Debug flow-specific traffic forwarding across the provider network.
- Analyze the behavior of various user-defined traffic patterns across the provider network.
- Perform the handover testing after initial deployment.
- Perform on-demand testing and results monitoring using a central entity.

The OmniSwitch implementation of CPE Test Head supports the ability to run unidirectional, ingress tests. Test setup involves configuring one CPE switch as the generator and a remote switch as the analyzer.

The following diagram shows an example of an OmniSwitch CPE Test Head configuration:



CPE Test Head Example - Unidirectional, Ingress Test

In this example:

- 1** The CPE test is started first on the analyzer switch and then on the generator switch. The analyzer switch sends packets to the generator switch to learn the source.
- 2** A configurable amount of traffic is generated and counted on the ingress UNI port of the generator switch, as if the traffic was generated from a test head connected to the UNI port. This subjects the test traffic to the ingress UNI SAP profile policies.
- 3** Traffic is counted and sent out on the SAP NNI port. This subjects the test frames to the egress NNI QoS policies.
- 4** Test frames are forwarded through the provider network over the customer EVC to the ingress NNI on the analyzer switch, where the packets are received and counted. Note that test frames are dropped after they are counted.

5 CPE Test Head CLI **show** commands are used on the generator and analyzer switches to display and verify test statistics, such as packets transmitted and received.

Note. The CPE test is non-disruptive to traffic running on other UNI ports that are associated with the same SAP profile as the test UNI port. All UNI and NNI ports, including CPE test ports, are subject to any SAP profile or QoS configuration associated with the SAP profile. This is important to consider when analyzing test results.

CPE Test Head Configuration Overview

CPE Test Head configuration is done using a test profile to define test attributes. Configuring a test profile is required on both the generator and analyzer switch. Not all test profile information is required for both switches. For example, the profile on the generator switch must contain a port number to identify the UNI port on which the test will run, but a port number is not required for the analyzer profile.

The following table provides a list of test profile parameters and identifies if the parameter is required on the generator, analyzer, or both. Also included is the CLI command used to configure the parameter.

Test Profile Parameters	Generator Switch	Analyzer Switch	CLI Command
Profile name	Yes	Yes	test-oam
Source and destination endpoints	Yes	Yes	test-oam src-endpoint dst-endpoint
Test frame source and destination MAC addresses	Yes	Yes	test-oam vlan test-frame
Service VLAN	Yes	Yes	test-oam vlan test-frame
Test role (generator or analyzer)	Yes	Yes	test-oam role
UNI port for test packet generation	Yes	No	test-oam port
Test frame parameters, such as VLAN tag, priority, and frame type	Yes	No	test-oam frame
Test duration, rate, and packet size	Yes	No	test-oam duration rate packet-size

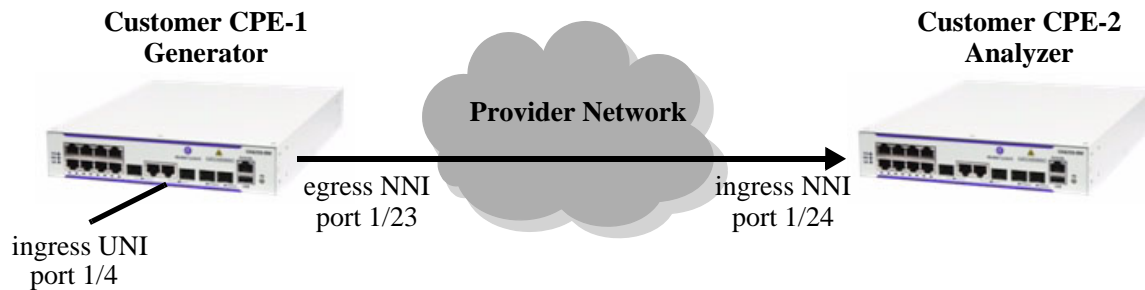
Configuration Guidelines

Consider the following guidelines when configuring the OmniSwitch CPE Test Head:

- Make sure the same test profile name (test ID) is used on the generator and analyzer switch.
- A switch can only perform one role (generator or analyzer) for a specific test.
- Only one test can be active for the switch at any given time.
- Up to 32 test profiles are allowed per switch.
- Regular traffic is disrupted on the ingress UNI port that is used to generate the test traffic. However, traffic on other UNI ports associated with the same SAP profile is not disrupted. Therefore, running the test on a UNI port that is not in use is recommended.

Configuring a CPE Test Profile

This section describes how to configure the following CPE test head example, which includes defining the test profile on the generator and analyzer switch. The configuration steps described in this section also provide a tutorial for how to use the OmniSwitch CLI to configure a CPE test.



To configure the test setup in the above example:

- 1 Configure the test profile name and an optional description on the generator (CPE-1 switch) and analyzer (CPE-2 switch) using the `test-oam` command. For example:

```
-> test-oam 100M_L2 descr "60 sec 100MB L2 test"
```

When the “100M_L2” test is created, a profile associated with this name is automatically created. This initial profile contains default parameter settings, where applicable. However, in some cases the default values are set to zero as a placeholder, but these parameters require additional configuration.

- 2 Configure the source (generator) and destination (analyzer) endpoints on CPE-1 and CPE-2 using the `test-oam src-endpoint dst-endpoint` command. For example:

```
-> test-oam 100M_L2 src-endpoint "CPE-1" dst-endpoint "CPE-2"
```

The endpoint is identified using the DNS host name for the switch. In this example, “CPE-1” and “CPE-2” are the configured host names for the generator and analyze switch.

- 3 Configure the service VLAN and the source and destination MAC for the test frame on CPE-1 and CPE-2 using the `test-oam vlan test-frame` command. For example:

```
-> test-oam 100M_L2 vlan 100 test-frame src-mac 00:00:00:11:11:11 dst-mac
00:00:00:22:22:22
```

- 4 Configure CPE-1 as the generator switch using the `test-oam role` command. For example:

```
-> test-oam 100M_L2 role generator
```

Use this command with the **generator** option on the CPE-1 switch. This will configure the role parameter in the “100M_L2” test profile that resides on CPE-1.

- 5 Configure CPE-2 as the analyzer switch using the `test-oam role` command. For example:

```
-> test-oam 100M_L2 role analyzer
```

Use this command with the **analyzer** option on the CPE-2 switch. This will configure the role parameter in the “100M_L2” test profile that resides on CPE-2.

Note that a switch can only serve as the generator or the analyzer for any given test.

- 6** Configure port 1/4 on CPE-1 as the port on which the test is run, using the **test-oam port** command. For example:

```
-> test-oam 100M_L2 port 1/4
```

This is the ingress UNI port that will generate test packets. The packets are then subject to the SAP profile and QoS policies that are associated with the port.

- 7** Configure the test duration, rate, and size of the test packet on CPE-1 using the **test-oam duration rate packet-size** command. For example:

```
-> test-oam 100M_L2 duration 100 rate 100m packet-size 1518
```

The test duration is the length of time, in seconds, that the test will run. The rate determines the rate at which packets are generated, in kbps or mbps. The packet size specifies the size of the test packet that is generated.

- 8** Configure a Layer 2 or Layer 3 test frame on CPE-1 using the **test-oam frame** command. The type of test needed determines the type of frame that is configured for the test. If a Layer 2 test is required, configure a Layer 2 frame type; if a Layer 3 test is required, configure a Layer 2 frame type. For example:

To configure a Layer 2 test frame, specify a hexadecimal value for the Ether type.

```
-> test-oam 100M_L2 frame vlan-tag 20 priority 5 ether-type 0x8101 data-pattern 0xabcd
```

To configure a Layer 3 test frame, specify the **ipv4** keyword for the Ether type.

```
-> test-oam 100M_IP frame vlan-tag 10 priority 5 ether-type ipv4 src-ip 10.10.10.111 dst-ip 10.10.10.222
```

See the **test-oam frame** command page in the *OmniSwitch 6250 CLI Reference Guide* for frame type parameter requirements and definitions.

The following provides a summary of the CLI commands used in the configuration example:

CPE-1 Generator	CPE-2 Analyzer
test-oam 100M_L2 descr "60 sec 100MB L2 Test"	test-oam 100M_L2 descr "60 sec 100MB L2 Test"
test-oam 100M_L2 src-endpoint CPE-1 dst-endpoint CPE-2	test-oam 100M_L2 src-endpoint CPE-1 dst-endpoint CPE-2
test-oam 100M_L2 vlan 100 test-frame src-mac 00:00:00:11:11:11 dst-mac 00:00:00:22:22:22	test-oam 100M_L2 vlan 100 test-frame src-mac 00:00:00:11:11:11 dst-mac 00:00:00:22:22:22
test-oam 100M_L2 role generator	test-oam 100M_L2 role analyzer
test-oam 100M_L2 port 1/4	
test-oam 100M_L2 duration 100 rate 100m packet-size 1518	
test-oam 100M_L2 frame vlan-tag 20 priority 5 ether-type 0x8101 data-pattern 0xabcd	

Refer to the *OmniSwitch 6250 CLI Reference Guide* for more information about these commands.

Running a CPE Test

A CPE test is started first on the analyzer switch and then on the generator switch using the **start** form of the **test-oam start stop** command. For example:

```
-> test-oam 100M_L2 start
```

This command also includes the following optional parameters used to specify runtime (active) values for the specified test:

- **vlan**—the service VLAN to use for the test.
- **port**—the port on which the test will generate test frames.
- **packet-size**—the size of the test frame to transmit.

When one or more of these runtime parameters are specified with the **test-oam start** command, the parameter value is used instead of the value configured for the same parameter in the CPE test profile. For example, if the “100M_L2” profile specifies port 1/10 for the test, the following command will run the “100M_L2” test on port 1/4:

```
-> test-oam 100M_L2 port 1/4 start
```

Note. The runtime values specified for any of the optional **test-oam start** command parameters do not overwrite the configured values for the test profile. In addition, if there are no configured values for these parameters in the profile and a runtime value is not specified with the command, the test will not run.

Stopping the CPE Test

An active CPE test is stopped when one of the following two actions occur:

- The duration time configured for the test profile is reached.
- The operator uses the **stop** form of the **test-oam start stop** command. For example:

```
-> test-oam 100M_L2 stop
```

Stopping the CPE test on both the generator and analyzer is recommended. The analyzer switch may continue to send out packets attempting to learn the test source if the test is not stopped on the analyzer switch as well.

Verifying the CPE Test Configuration and Results

To display the CPE test configuration and statistics information, use the **show** commands listed below:

- show test-oam** Displays the test configuration and status.
show test-oam statistics Displays test statistics.

The **show test-oam** command displays a summary of CPE test information or more detailed information for a specific test. For example:

```
-> show test-oam tests
Total Test-Ids: 4
Test-Id  Port  Src-Mac          Dst-Mac          Vlan  Direction  Status
-----+-----+-----+-----+-----+-----+-----
Test1    1/1  00:11:22:33:44:55  00:22:33:44:55:66  100   unidirectional  ended
Test2    1/2  00:44:22:33:44:55  00:66:33:44:55:66  200   unidirectional  stopped
Test3    2/3  00:00:00:00:00:03  00:00:00:00:00:04  200   unidirectional  not-started
Test4    1/1  00:00:00:00:00:07  00:00:00:00:00:08  100   unidirectional  running
```

```
-> show test-oam Test1
Legend: dei-drop eligible indicator
TEST Parameters for Test1:
  Source Endpoint      : SW1,
  Destination Endpoint : SW2,
  Test Description     : Ether Test,
  Direction            : unidirectional,
  Source MAC           : 00:11:22:33:44:55,
  Destination MAC      : 00:22:33:44:55:66,
  Duration             : 10(secs),
  Vlan                 : 100,
  Role                 : generator,
  Port                 : 1/1,
  Tx Rate              : 80m,
  Frame Size           : 100,
  State                : start,
  Status               : running
```

```
Frame Configuration:
  Frame Type : ether,
  Vlan       : 200,
  Priority   : 7,
  Pattern    : 0x0001,
  Dei        : none,
  Ether Type : 0x8000,
```

The **show test-oam statistics** command displays packet counts for the number of test packets transmitted and received. For example:

```
-> show test-oam statistics
Test-Id      TX-Ingress  TX-Egress  RX-Ingress
-----+-----+-----+-----
Test1        1200366    1200366    0
Test2         0          0          1200366
```

The packet counts displayed are based on the role the switch plays for the specific test. For example, "Test1" shows statistics for **TX-Ingress** (packets transmitted on ingress UNI) and **TX-Egress** (packets transmitted on egress NNI), but not for **RX-Ingress** (packets received on ingress NNI). This is because the

show command was performed on the generator switch for “Test1”. The “Test2” display output only shows statistics for **RX-Ingress** because the switch is the analyzer for “Test2”.

To verify the received test packet count for “Test1”, use the **show test-oam statistics** command on the analyzer switch. To verify the transmitted test packet count for “Test2”, use the same **show** command on the generator switch.

Note. For more information about the resulting display from these commands, see the *OmniSwitch CLI Reference Guide*.

15 Configuring Ethernet OAM

The rise in the number of Ethernet service instances has resulted in service providers requiring a powerful and robust set of management tools to maintain Ethernet service networks. Service provider networks are large and intricate, often comprising of different operators that work together to provide the customers with end-to-end services. The challenge for the service providers is to provide a highly available convergent network to its customer base. Ethernet OAM (Operations, Administration, and Maintenance) provides the detection, resiliency, and monitoring capability for end-to-end service guarantee in an Ethernet network.

In This Chapter

This chapter describes the Ethernet OAM feature, how to configure it and display Ethernet OAM information through the Command Line Interface (CLI). For more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

The following procedures are described in this chapter:

- [“Ethernet OAM Overview” on page 15-3.](#)
- [“Elements of Service OAM” on page 15-3.](#)
- [“Fault Management” on page 15-5.](#)
- [“Performance Monitoring” on page 15-5.](#)
- [“Interoperability with ITU-T Y.1731” on page 15-7.](#)
- [“Configuring Ethernet OAM” on page 15-9.](#)
- [“Verifying the Ethernet OAM Configuration” on page 15-14.](#)

Ethernet OAM Specifications

The following table lists Ethernet OAM specifications.

Standards Supported	IEEE 802.1ag Version 8.1– <i>Connectivity Fault Management</i> IEEE 802.1D– <i>Media Access Control (MAC) Bridges</i> IEEE 802.1Q– <i>Virtual Bridged Local Area Networks</i> ITU-T Y.1731– <i>OAM Functions and Mechanisms for Ethernet-Based Networks</i>
Platforms Supported	OmniSwitch 6250-Metro Models
Maximum Maintenance Domains (MD) per Bridge	4
Maximum Maintenance Associations (MA) per Bridge	64
Maximum Maintenance End Points (MEP) per Bridge	128
Maximum MEP CMM Database Size	4092

Ethernet OAM Defaults

The following table shows Ethernet OAM default values.

Parameter Description	Command	Default Value/Comments
MHF value assigned to a MD	ethoam domain mhf	none
ID-permission value for MD entry	ethoam domain id-permission	none
MHF value assigned to a MA	ethoam association mhf	defer
Continuity Check Message interval for the MA	ethoam association ccm-interval	10 seconds
Default domain level	ethoam default-domain level	0
Default domain MHF value	ethoam domain mhf	none
Default domain ID permission	ethoam default-domain id-permission	none
The administrative status of the MEP	ethoam endpoint admin-state	disable
The priority value for CCMs and LTMs transmitted by the MEP	ethoam endpoint priority	7
The lowest priority fault alarm for the lowest priority defect for a MEP	ethoam endpoint lowest-priority-defect	mac-rem-err-xcon
Number of Loopback messages	ethoam loopback	1
Fault notification alarm time	ethoam fault-alarm-time	250 centiseconds
Fault notification generation reset time	ethoam fault-reset-time	1000 centiseconds

Ethernet OAM Overview

Ethernet OAM focuses on two main areas that service providers require the most and are rapidly evolving in the standards bodies:

- Service OAM (IEEE 802.1ag and ITU-T Y.1731)—for monitoring and troubleshooting end-to-end Ethernet service instances.
- Link OAM (IEEE 802.3ah EFM Link OAM)—for monitoring and troubleshooting individual Ethernet links.

These two protocols are both unique and complimentary. For example, Service OAM may isolate a fault down to a specific service, but to determine exactly where the fault occurred within the network infrastructure might also require the use of Link OAM.

This chapter provides information about configuring Service OAM. For information about Link OAM, see [Chapter 17, “Configuring EFM \(LINK OAM\).”](#)

Ethernet Service OAM

Ethernet Service OAM Connectivity Fault Management (CFM) allows service providers to manage customer services end-to-end on a per-service-instance basis. A customer service instance, or Ethernet Virtual Connection (EVC), is the service that is sold to a customer and is designated by a VLAN tag on the User-to-Network Interface (UNI).

Elements of Service OAM

- Maintenance End Points (MEPs) and Maintenance Intermediate Points (MIPs)
 - MEPs initiate OAM commands. MEPs prevent leakage between domains.
 - MIPs passively receive and respond to OAM frames.
- Maintenance Association (MA) is a logical connection between 2 or more MEPs.
- Point-to-point MA: logical sub-MA component only between 2 MEPs MA.
- Maintenance Domain (MD): One or more MAs under the same administrative control.
- Maintenance Domain Levels: There are 8 levels defined in 802.1ag:
 - levels [5, 6, 7] are for operators,
 - levels [3, 4] are for service provider
 - levels [0, 1, 2] are for customersMultiple levels are supported for flexibility.
- Mechanisms: continuity check (CC), loopback, link trace

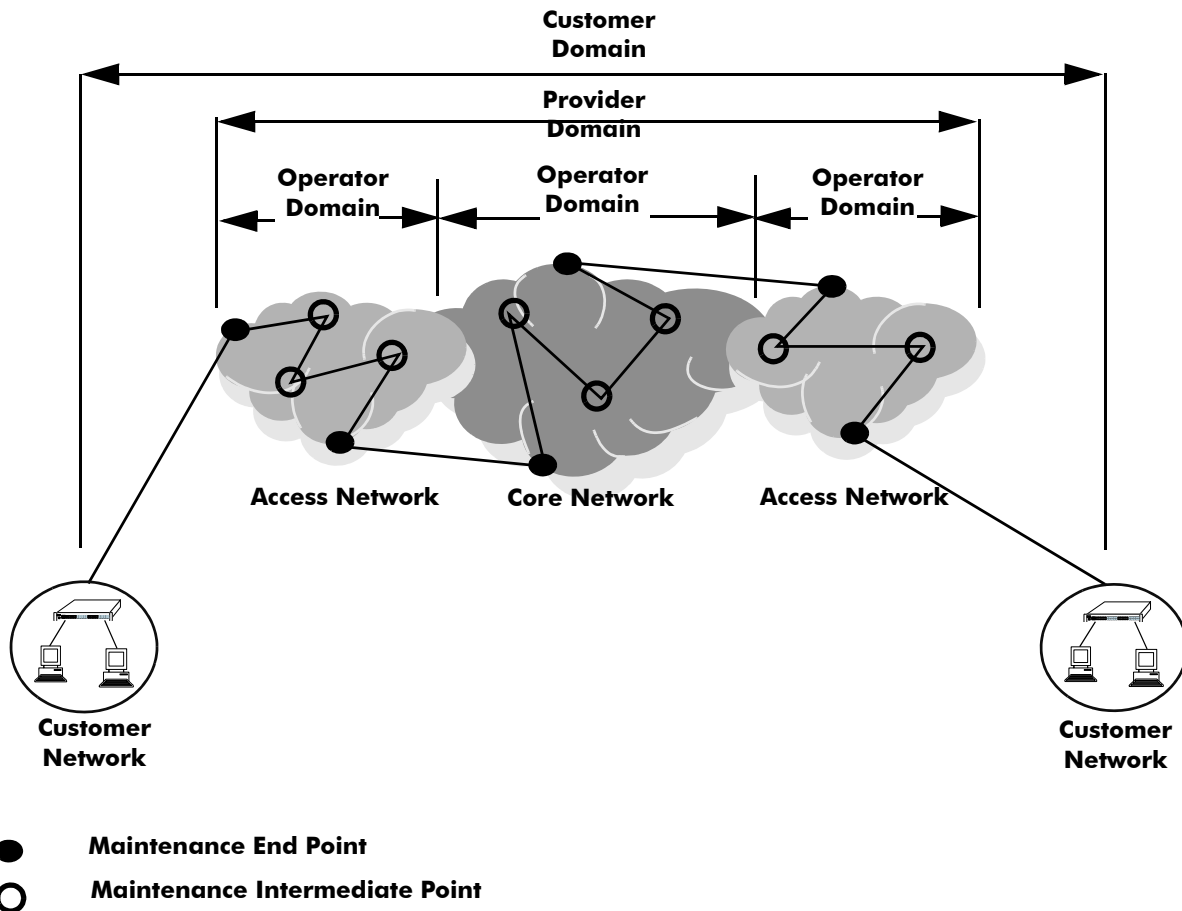
CFM Maintenance Domain

CFM uses a hierarchical Maintenance Domain (MD) infrastructure to manage and administer Ethernet networks.

- Each domain is made up of Maintenance Endpoints (MEPs) and Maintenance Intermediate Points (MIPs).

- The MEPs are configured on edge ports within the domain for each EVC. The MIPs are configured on relevant ports within the domain itself (interior ports).
- The network administrator selects the relevant points within the network to determine where maintenance points are needed. The maintenance point configuration defines the MD.
- MDs are assigned a unique level number (between 0 and 7) to help identify and differentiate the MD within the domain hierarchy. For example, different organizations, such as customers (levels 0, 1, 2), service providers (levels 3, 4), and operators (levels 5, 6, 7), are involved in a Metro Ethernet Service.
- Each organization can have its own Maintenance Domain, designated by the assigned level number to specify the scope of management needed for that domain.

The following illustration shows an example of the CFM Maintenance Domain hierarchy:



CFM Maintenance Domain Hierarchy

Fault Management

Service OAM Connectivity Fault Management consists of three types of messages that are used to help network administrators detect, verify, and isolate when a problem occurs in the network:

- **Continuity Check Messages (CCM)**—These are multicast messages exchanged periodically by MEPs to detect loss of service connectivity between MEPs. These messages are also used by MEPs and MIPs to discover other MEPs within a domain.
- **Linktrace Messages (LTM)**—These messages are transmitted by a MEP to trace the path to a destination maintenance point. The receiving maintenance point responds to LTMs with a linktrace reply (LTR). This mechanism is similar to the UDP Trace Route function. The transmission of linktrace messages is requested by an administrator.
- **Loopback Messages (LBM)**—These messages are transmitted by a MEP to a specified MIP or MEP to determine whether or not the maintenance point is reachable. The receiving maintenance point responds to LBMs with a loopback reply (LBR). This mechanism is not used to discover a path to the destination; it is similar to the Ping function. The transmission of loopback messages is requested by an administrator.

MIP CCM Database Support

Per section 19.4 of the IEEE 802.1ag 5.2 draft standard, an MHF may optionally maintain a MIP CCM database as it is not required for conformance to this standard. A MIP CCM database, if present, maintains the information received from the MEPs in the MD and can be used by the Linktrace Protocol.

This implementation of Ethernet OAM does not support the optional MIP CCM database. As per section 19.4.4 of the IEEE 802.1ag 5.2 draft standard, LTM is forwarded on the basis of the source learning filtering database. Because the MIP CCM database is not supported in this release, MIPs will not forward LTM on blocked egress ports.

Performance Monitoring

The ITU-T Y.1731 Recommendation addresses the need to monitor performance to help enforce customer service level agreements (SLAs). Frame delay (latency) and frame delay variation (jitter) are important performance objectives, especially for those applications (such as voice) that cannot function with a high level of latency or jitter.

This implementation of Service OAM supports Ethernet frame delay measurement (ETH-DM) and is compliant with Y.1731. The ETH-DM feature allows for the configuration of on-demand OAM to measure frame delay and frame delay variation between endpoints.

Frame delay measurement is performed between peer MEPs (measurements to MIPs are not done) within the same MA. Although the OmniSwitch implementation of ETH-DM is compliant with ITU-T Y.1731, delay measurement can be performed for both ITU-T Y.1731 and IEEE 802.1ag MEPs.

Any MEP can initiate or reply to an ETH-DM request, depending on the type of delay measurement requested. There are two types of delay measurements supported: one-way and two-way.

One-way ETH-DM

- A MEP sends one-way delay measurement (1DM) frames to a peer MEP. The sending MEP inserts the transmission time into the 1DM frame at the time the frame is sent.
- When a MEP receives a 1DM frame, the MEP calculates the one-way delay as the difference between the time at which the frame was received and the transmission time indicated by the frame timestamp (receive time minus transmission time).

- One-way delay measurement statistics are gathered and stored on the receiving MEP (the MEP that receives a 1DM request).
- One-way ETH-DM requires clock synchronization between the sending and receiving MEPs. Using NTP for clock synchronization is recommended.

Two-way ETH-DM

- A MEP sends delay measurement message (DMM) frames to a peer MEP to request a two-way ETH-DM. The sending MEP inserts the transmission time into the DMM frame at the time the frame is sent.
- When a MEP receives a DMM frame, the MEP responds to the DMM with a delay message reply (DMR) frame that contains the following timestamps:
 - Timestamp copied from the DMM frame.
 - Timestamp indicating when the DMM frame was received.
 - Timestamp indicating the time at which the receiving MEP transmitted the DMR frame back to the sending MEP.
- When a MEP receives a DMR frame, the MEP compares all the DMR timestamps with the time at which the MEP received the DMR frame to calculate the two-way delay.
- The two-way delay is the difference between the time the originating MEP sent a DMM request and the time at which the originating MEP received a DMR frame minus the time taken by the responding MEP to process the DMM request.
- Two-way delay measurement statistics are gathered and stored on the originating MEP (the MEP that initiates a DMM request).
- This method *does not* require clock synchronization between the transmitting and receiving MEPs.
- Two-way ETH-DM is an on-demand OAM performance measurement. To set up continuous two-way delay measurement, see the “Service Assurance Agent Commands” chapter in the *OmniSwitch CLI Reference Guide* for information about how to configure a SAA for continuous two-way frame delay measurement.

Frame Delay Variation

The delay variation (jitter) for both one-way and two-way ETH-DM is determined by calculating the difference between the current delay measurement value and the previous delay measurement value. If a previous delay value is not available, which is the case when a DM request is first made, then jitter is not calculated.

Interoperability with ITU-T Y.1731

This implementation of Ethernet Service OAM supports both IEEE 802.1ag and ITU-T Y.1731 for connectivity fault management (plus performance monitoring provided by ITU-T Y.1731). Although both standards are supported, the OmniSwitch implementation uses the 802.1ag terminology and hierarchy for Ethernet CFM configuration.

The following table provides a mapping of 802.1ag terms to the equivalent ITU-T Y.1731 terms:

IEEE 802.1ag v8.1	ITU-T Y.1731
Maintenance Domain (MD)	Maintenance Entity (ME)
Maintenance Association (MA)	Maintenance Entity Group (MEG)
Maintenance Endpoint (MEP)	MEG Endpoint (MEP)
Maintenance Intermediate Point (MIP)	MEG Intermediate Point (MIP)
Maintenance Domain Level	MEG Level

Support for both the IEEE and ITU-T Ethernet CFM standards allows interoperability between OmniSwitch 802.1ag and Y.1731 CFM with the following minor configuration requirements:

- The OmniSwitch MD format must be configured as “none”.
- ITU-T Y.1731 uses the “icc-based” format for a MEG, so the OmniSwitch MA format must also be configured to use the “icc-based” format.
- When the OmniSwitch MA is configured with the “icc-based” format, the MA name is automatically padded with zeros if the name specified is less than 13 characters.

The OmniSwitch CLI commands to configure an MD and MA include the “none” and “icc-based” format options. See [“Configuring Ethernet OAM” on page 15-9](#) for more information.

Quick Steps for Configuring Service OAM

The following steps provide a quick tutorial on how to configure Ethernet OAM. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

- 1 Create an Ethernet domain using the **ethoam domain** command. For example:

```
-> ethoam domain esd.alcatel-lucent.com format dnsName level 1
```

- 2 Create an Ethernet OAM Maintenance Association using the **ethoam association** command. For example:

```
-> ethoam association alcatel-sales format string domain esd.alcatel-lucent.com  
vlan 10
```

- 3 Create an Ethernet OAM Maintenance End Point using the **ethoam endpoint admin-state** command. For example:

```
-> ethoam endpoint 100 domain esd.alcatel-lucent.com association alcatel-sales  
direction up port 1/10
```

- 4 Administratively enable the Ethernet OAM Maintenance End Point using the **ethoam endpoint admin-state** command. For example:

```
-> ethoam endpoint 100 domain esd.alcatel-lucent.com association alcatel-sales  
admin-state enable
```

- 5 Enable Continuity Check Messages for the Ethernet OAM Maintenance End Point using the **ethoam endpoint ccm** command. For example:

```
-> ethoam endpoint 100 domain esd.alcatel-lucent.com association alcatel-sales  
ccm enable
```

- 6 Configure the Message Handling Function (MHF) value of an Ethernet OAM Maintenance Domain using the **ethoam domain mhf** command. For example:

```
-> ethoam domain esd.alcatel-lucent.com mhf explicit
```

- 7 Configure the endpoint list for the Ethernet OAM Maintenance Association using the **ethoam association endpoint-list** command. For example:

```
-> ethoam association alcatel-sales domain esd.alcatel-lucent.com endpoint-list  
100
```

- 8 Enable the maintenance entity to initiate transmitting loopback messages to obtain loopback replies using the **ethoam loopback** command. For example:

```
-> ethoam loopback target-endpoint 15 source-endpoint 100 domain esd.alcatel-  
lucent.com association alcatel-sales
```


Configuring Ethernet OAM

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to configure Ethernet Service OAM on a switch. Consider the following guidelines when configuring Service OAM maintenance entities:

- Ethernet OAM is not supported on mobile, mirrored, or aggregate ports (the physical port members of an aggregate).
- Ethernet OAM is also not supported on dynamically learned VLANs.
- Implementing Ethernet OAM is supported on any full-duplex point-to-point or emulated point-to-point Ethernet link. It need not be implemented system wide.
- Management systems are important for configuring Ethernet OAM across the network. They also help to automate network monitoring and troubleshooting. Ethernet OAM can be configured in two phases: network configuration phase and service activation phase.
- The network configuration phase enables Connectivity Fault Management (CFM) on the switches. This is also the phase where Maintenance Intermediate Points (MIP) and Maintenance End Points (MEP) are identified and set up.
- Any port on a switch is referred to as a Maintenance Point (MP). An MP can be either a MEP or MIP. A MEP resides at the edge of a Maintenance Domain (MD), while a MIP is located within a MD.
- In the Service Activation phase, a new end point is created on a VLAN as a MEP. This enables the configuration of continuity-check and cross-check functionality.

Configuring a Maintenance Domain

To create a Maintenance Domain (MD), use the **ethoam domain** command, by entering **ethoam domain**, followed by the domain name, the keyword **format**, the domain name format type, the keyword **level**, and the level of the domain. For example:

```
-> ethoam domain esd.alcatel-lucent.com format dnsName level 5
```

Here, the MD **esd.alcatel.com** is created.

Note that the level must be 0-2 at operator level, 3-5 at provider level, and 6-7 at customer level when creating the level of domain.

To remove an MD, use the **no** form of this command. For example:

```
-> no ethoam domain esd.alcatel-lucent.com
```

Note that with this implementation of Ethernet OAM, it is only possible to delete an MD when there is no Maintenance Association, End Point, or Intermediate Point associated with the MD.

Modifying a Maintenance Domain

To modify the MHF value of an MD, use the **ethoam domain mhf** command, as shown:

```
-> ethoam domain esd.alcatel-lucent.com mhf explicit
```

To modify the default Ethernet OAM Maintenance Domain, use the **ethoam default-domain level** command, as shown:

```
-> ethoam default-domain vlan 100 level 4 mhf none
```

Note. The **no** form of this command restores the default Ethernet OAM Maintenance Domain value.

Configuring a Maintenance Association

To create an Ethernet OAM Maintenance Association (MA), use the **ethoam association** command. For example, to create the MA **alcatel-sales** in the **esd.alcatel.com** domain, enter:

```
-> ethoam association alcatel-sales format string domain esd.alcatel-lucent.com  
primary-vlan 10
```

To remove an MA, use the **no** form of this command. For example:

```
-> no ethoam association alcatel-sales domain esd.alcatel-lucent.com
```

Note that with this implementation of Ethernet OAM, it is only possible to delete an MA when there is no Maintenance End Point (MEP) or Maintenance Intermediate Point (MIP) associated with the MA.

Configuring Maintenance Association Attributes

The MIP Half Function (MHF), Continuity Check Message (CCM) interval, and MEP list are configurable attributes of a Maintenance Association.

By default, the MHF value is set to defer. To modify this value for an MA, use the **ethoam association mhf** command. For example:

```
-> ethoam association alcatel-sales domain esd.alcatel-lucent.com mhf default
```

By default, the CCM interval is set to 10 seconds. To modify this value for an MA, use the **ethoam association ccm-interval** command:

```
-> ethoam association alcatel-sales domain esd.alcatel-lucent.com ccm-interval  
intervallm
```

To modify the MEP list of an MA, use the **ethoam association endpoint-list** command, as shown:

```
-> ethoam association alcatel-sales domain esd.alcatel-lucent.com endpoint-list  
100-200
```

To remove the MEP list from an Ethernet OAM Maintenance Association, enter:

```
-> no ethoam association alcatel-sales domain esd.alcatel-lucent.com endpoint-  
list 100-200
```

Configuring a Maintenance End Point

To create an Ethernet OAM Maintenance End Point (MEP), use the **ethoam endpoint** command. For example, to create UP MEP 100 in domain “esd.alcatel-lucent.com” of the “alcatel-sales” Maintenance Association on port 1/2 of VLAN 400, enter:

```
-> ethoam end-point 100 domain esd.alcatel-lucent.com association alcatel-sales
direction up port 1/2 vlan 400
```

To remove a MEP, use the **no** form of this command. For example:

```
-> no ethoam end-point 100 domain esd.alcatel-lucent.com association alcatel-
sales
```

To configure the administrative state of a MEP, use the **ethoam endpoint admin-state** command. For example:

```
-> ethoam end-point 100 domain esd.alcatel-lucent.com association alcatel-sales
admin-state enable
```

Configuring MEP Attributes

To configure the MEP to generate Continuity Check Messages (CCM), use the **ethoam endpoint ccm** command. For example:

```
-> ethoam end-point 100 domain esd.alcatel-lucent.com association alcatel-sales
ccm enable
```

To configure the priority values for Continuity Check Messages and Linktrace Messages transmitted by a MEP, use the **ethoam endpoint priority** command. For example:

```
-> ethoam end-point 100 domain esd.alcatel-lucent.com association alcatel-sales
priority 6
```

To configure the lowest priority fault alarm for the lowest priority defect for a MEP, use the **ethoam endpoint lowest-priority-defect** command. For example:

```
-> ethoam end-point 100 domain esd.alcatel-lucent.com association alcatel-sales
lowest-priority-defect all-defect
```

Configuring Loopback

To initiate transmitting Loopback messages (LBMs) and obtaining Loopback replies (LBRs), use the **ethoam loopback** command. For example:

```
-> ethoam loopback target-endpoint 10 source-endpoint 20 domain MD association
MA number 3
Reply from 00:0E:B1:6B:43:89: bytes=64 seq=0 time=100ms
Reply form 00:0E:B1:6B:43:89: bytes=64 seq=0 time=112ms
Request timed out.
----00:E0:B1:6B:43:89 ETH-LB Statistics----
3 packets transmitted, 2 packets received, 33% packet loss
round-trip (ms) min/avg/max = 100/106/112
```

Configuring Linktrace

To initiate transmitting Linktrace messages (LTMs) and detecting Linktrace replies (LTR), use the **ethoam linktrace** command. For example:

```
-> ethoam linktrace 10:aa:ac:12:12:ad end-point 4 domain esd.alcatel-lucent.com
association alcatel_sales flag fdonly hop-count 32
```

Configuring the Fault Alarm Time

The Fault Alarm time is the period of time during which one or more defects should be detected before the Fault Alarm is issued. By default, this timer is set to 250 centiseconds. To change the Fault Alarm time, use the **ethoam fault-alarm-time** command. For example:

```
-> ethoam fault-alarm-time 500 end-point 100 domain esd.alcatel-lucent.com asso-
ciation alcatel_sales
```

Configuring the Fault Reset Time

The Fault Reset time is the time interval in which Fault Alarm is re-enabled to process the faults. By default, this timer value is set to 1000 centiseconds. To change the Fault Reset time, use the **ethoam fault-reset-time** command. For example:

```
-> ethoam fault-reset-time 250 end-point 100 domain esd.alcatel-lucent.com asso-
ciation alcatel_sales
```

Configuring Ethernet Frame Delay Measurement

Ethernet frame delay measurement (ETH-DM) is an on-demand OAM function used to measure frame delay (latency) and delay variation (jitter) between MEPs. There are two types of ETH-DM supported: one-way and two-way.

One-Way ETH-DM

The **ethoam one-way-delay** command is used to configure a one-way ETH-DM (1DM) to monitor performance between two MEPs. For example, the following command is used to initiate the transmission of 1DM frames to a target MEP:

```
-> ethoam one-way-delay target-endpoint 10 source-endpoint 12 domain MD1 associ-
ation MA1 vlan-priority 4
```

This command initiates the sending of 1DM frames from MEP 12 to MEP 10, which does not reply to frames received from MEP 12. The latency and jitter statistics are gathered and stored on the receiving MEP, which is MEP 10 in this example.

An option to specify a target MAC address, instead of a MEP ID, is also supported. For example:

```
-> ethoam one-way-delay target-macaddress 00:e0:b1:6a:52:4c source-endpoint 12
domain MD association MA vlan-priority 4
```

One-way delay measurement statistics are gathered and stored on the receiving MEP (the MEP that receives a 1DM request).

Note. One-way ETH-DM requires clock synchronization between the sending and receiving MEPs. Using NTP for clock synchronization is recommended.

Two-Way ETH-DM

The `ethoam two-way-delay` command is used to configure a two-way ETH-DM to monitor roundtrip performance between two MEPs. For example, the following command is used to initiate the transmission of delay measurement message (DMM) frames to a target MEP:

```
-> ethoam two-way-delay target-endpoint 10 source-endpoint 12 domain MD associa-
tion MA vlan-priority 4
Reply from 00:0E:B1:6B:43:89 delay=2584us jitter=282us
```

This command initiates the sending of DMM frames from MEP 12 to MEP 10. However, with two-way delay measurement, the receiving MEP replies with delay message response (DMR) frames to the sending MEP. In this example, MEP 10 sends DMR frames back to MEP 12.

An option to specify a target MAC address, instead of a MEP ID, is also supported. For example:

```
-> ethoam two-way-delay target-macaddress 00:e0:b1:6a:52:4c source-endpoint 12
domain MD association MA vlan-priority 4
Reply from 00:E0:B1:6A:52:4C: delay=2584us jitter=282us
```

Note the following when configuring two-way ETH-DM:

- Two-way delay measurement statistics are gathered and stored on the originating MEP (the MEP that initiates a DMM request).
- This method *does not* require clock synchronization between the transmitting and receiving MEPs.
- Two-way ETH-DM is an on-demand OAM performance measurement. To schedule continuous two-way delay measurement, see [Chapter 16, “Service Assurance Agents \(SAA\),”](#) for more information.

Verifying the Ethernet OAM Configuration

To display information about Ethernet OAM on the switch, use the show commands listed below:

show ethoam	Displays the information of all the Management Domains configured on the switch.
show ethoam domain	Displays the information of a specific Management Domain configured on the switch.
show ethoam domain association	Displays the information of a specific MA in a Management Domain configured on the switch.
show ethoam domain association end-point	Displays the information of a specific MEP in a Management Domain configured on the switch.
show ethoam default-domain	Displays all the default MD information for all the VLANs or a specific VLAN.
show ethoam remote-endpoint	Displays the information of all remote MEPs learned as a part of the CCM message exchange.
show ethoam cfmstack	Displays the contents of CFM Stack Managed Object, which determines the relationships among MEPs and MIPs on a specific switch port.
show ethoam linktrace-reply	Displays the content of the Linktrace reply (LTR) returned by a previously transmitted LTM. This command displays the LTR based on the transaction identifier or sequence number of the LTM for which the LTR is to be displayed
show ethoam linktrace-tran-id	Displays the transaction identifiers returned by previously generated LTMs from a specified MEP.
show ethoam vlan	Displays the Ethernet OAM statistics of all the Management Domains configured on the switch. Also, displays the statistics of all the MAs and matching MEPs for all the MDs.

16 Service Assurance Agents (SAA)

With SAAs, users can verify service guarantees, increase network reliability by validating network performance, proactively identify network issues, and increase Return on Investment (ROI) by easing the deployment of new services. SAA uses active monitoring to generate traffic in a continuous, reliable, and predictable manner, thus enabling the measurement of network performance and health.

IP SAAs enhance the service level monitoring to become IP application-aware by measuring both end-to-end and at the IP layer. IP SAA allows performance measurement against any IP addresses in the network (e.g., switch, server, PC). ETH-LB/DMM can be used to measure delay and jitter by sending out frames with DM information to the peer MEP and receiving frames with DM information from the peer MEP..

In This Chapter

This chapter describes the various types of SAAs that can be configured on an OmniSwitch. Configuration procedures described in this chapter include:

Service Assurance Agent

- Configuring SAA for MAC Address on [page 16-4](#).
- Configuring SAA for IP on [page 16-4](#).
- Configuring SAA for Ethoam Loopback on [page 16-4](#).
- Configuring SAA for ETH-DMM on [page 16-4](#).
- Displaying SAA Configuration on [page 16-5](#).

SAA Specifications

The following table lists Ethernet OAM specifications.

IEEE Standards Supported	N/A
Platforms Supported	OmniSwitch 6250-Metro Models

SAA Defaults

The following table shows SAA default values.

Parameter Description	Command	Default Value/Comments
Configure SAA for ETH-LB	<code>saa type ethoam-loopback</code>	5
Configure SAA for ETH-DMM	<code>saa type ethoam-two-way-delay</code>	5

Quick Steps for Configuring SAA

The following steps provide a quick tutorial on how to configure SAA. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

- 1 Configure SAA for IP using the **saa type ip-ping** command. For example:

```
-> saa "saa-ip" type ip-ping destination-ip 123.32.45.76 source-ip 123.35.42.124
type-of-service 4
```

- 2 Configure SAA for MAC using the **saa type mac-ping** command. For example:

```
-> saa "saa-mac" type mac-ping destination-macaddress 00:11:11:11:11:11 vlan 10
vlan-priority 3
```

- 3 Configure SAA for Ethoam loopback using the **saa type ethoam-loopback** command.

For example:

```
-> saa "saa-lb" type ethoam-loopback target-endpoint 10 source endpoint 1 domain
mdl association mal vlan-priority 5 drop-eligible false
```

- 4 Configure SAA for ETH-DMM using **saa type ethoam-two-way-delay** command. For example:

```
-> saa "saa-dmm" type ethoam-two-way-delay target-endpoint 10 source endpoint 1
domain mdl association mal vlan-priority 5
```

- 5 Start the saa using the **saa start** command.

```
-> saa "saa-ip" start
```

- 6 Stop the saa using the **saa stop** command.

```
-> saa "saa-ip" stop
```

Configuring Service Assurance Agent (SAA)

With SAAs, users can verify service guarantees, increase network reliability by validating network performance and proactively identify network issues. SAA uses active monitoring to generate traffic in a continuous, reliable, and predictable manner, thus enabling the measurement of network performance and health.

IP SAAs enhance the service level monitoring to become IP application-aware by measuring both end-to-end and at the IP layer. IP SAA allows performance measurements against any IP addresses in the network (e.g., switch, server, PC). ETH-LB/DMM can be used to measure delay and jitter by sending out frames with DM information to the peer MEP and receiving frames with DM information from the peer MEP.

Configuring SAA for MAC Addresses

L2 SAAs enhance the service level monitoring by enabling performance measurement against any L2 address within the provider network.

To configure SAA for MAC, use the **saa type mac-ping** command, by entering **saa**, followed saa name, keyword **type mac-ping**, keyword **destination-macaddress**, the destination MAC address as well any other additional parameters as shown below:

```
-> saa saa5 type mac-ping destination-macaddress 00:11:11:11:11:11 vlan 10
data "asdf" drop-eligible true vlan-priority 3 num-pkts 4
```

Configuring SAA for IP

To configure SAA for IP, use the **saa type ip-ping** command, by entering **saa**, followed saa name, keyword **type ip-ping**, keyword **destination-ip**, the destination ip address, keyword **source-ip**, the source ip address, the keyword **type-of-service** and type of service.

```
-> saa "saal" type ip-ping destination-ip 123.32.45.76 source-ip 123.35.42.124
type-of-service 4
```

Configuring SAA for Ethoam Loopback

To configure SAA for Ethoam Loopback, use the **saa type ethoam-loopback** command, by entering **saa**, followed saa name, keyword **type ethoam-loopback**, keyword **target-endpoint**, the id of destination endpoint, keyword **source-endpoint**, the id of source endpoint, the keyword **domain**, the domain name, the keyword **association**, the association name, the keyword **vlan-priority**, the vlan priority number, the keyword **drop-eligible** and drop-eligible value (true or false).

```
-> saa "saal" type ethoam-loopback target-endpoint 10 source endpoint 1 domain
mdl association mal vlan-priority 5 drop-eligible false
```

Configuring SAA for ETH-DMM

To configure SAA for ETH-DMM, use the **saa type ethoam-two-way-delay** command, by entering **saa**, followed saa name, keyword **type ethoam-two-way-delay**, keyword **target-endpoint**, the id of destination endpoint, keyword **source-endpoint**, the id of source endpoint, the keyword **association**, the association name, the keyword **vlan-priority**, the vlan priority number

```
-> saa "saal" type ethoam-two-way-delay target-endpoint 10 source endpoint 1
domain mdl association mal vlan-priority 5
```

Starting and Stopping SAAs

Once an SAA is configured it must be started and stopped using the **saa start** and **saa stop** commands as shown in the example below:

```
-> saa "saal" start
-> saa "saal" stop
```

Displaying the SAA Configuration

To display information about SAA on the switch, use the show commands listed below:

show saa	Displays generic configuration parameters of all the SAAs maintained at a given point of time.
show saa statistics	Displays SAA statistics.
show saa type config	Displays configured SAAs of the given type.

17 Configuring EFM (LINK OAM)

Ethernet in the First Mile (EFM), also known as LINK OAM, is a collection of protocols specified in IEEE 802.3ah, defining Ethernet in the access networks that connects subscribers to their immediate service provider. EFM, EFM-OAM and LINK OAM refers to IEEE 802.3ah standard.

LINK OAM (Operation, Administration, and Maintenance) is a tool monitoring Layer-2 link status by sending OAM protocol data units (OAMPDUs) between networked devices on the first mile. The first mile network refers to the connection between the subscriber and the public carrier network. LINK OAM is mainly used to address common link-related issues on the first mile. It helps network administrators manage their networks effectively.

By enabling LINK OAM on two devices connected by a point-to-point connection, network administrators can monitor the status of the link, detect faults in network segments, and probe link errors by using loopback testing.

In This Chapter

This chapter describes the LINK OAM feature and how to configure it through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*. This chapter provides an overview of LINK OAM and includes the following information:

- [“LINK OAM Specifications” on page 17-2](#)
- [“LINK OAM Defaults” on page 17-3](#)
- [“Quick Steps for Configuring LINK OAM” on page 17-4](#)
- [“Interaction With Other Features” on page 17-8](#)
- [“Configuring Link Monitoring” on page 17-10](#)
- [“Configuring LINK OAM” on page 17-9](#)
- [“Verifying the LINK OAM Configuration” on page 17-12](#)

LINK OAM Specifications

IEEE Standards Supported	IEEE 802.3ah– <i>EFM LINK OAM</i> RFC 4878 - <i>Definitions and Managed Objects for Operations, Administration, and Maintenance (OAM) functions on Ethernet-Like Interfaces.</i>
Platforms Supported	OmniSwitch 6250-Metro Models.
Maximum LINK OAM instances per switch	24 ports per NI and 48 ports per switch.
Maximum loopback sessions	2 simultaneous loopback sessions per NI.
Maximum event logs	64 most recent event logs is supported per port
Mirroring ports	LINK OAM is not supported on mirroring ports.

LINK OAM Defaults

The following table shows LINK OAM default values.

Parameter Description	Command	Default Value/Comments
Multiple PDU count assigned for event notifications.	efm-oam multiple-pdu-count	3
Maximum time period for which a LINK OAM port shall wait for a hello message from its peer before resetting a discovery session.	efm-oam port keepalive-interval	5 seconds
Time interval (in seconds) by which the information OAMPDUs are transmitted out of an LINK OAM enabled port.	efm-oam port hello-interval	1 second
Propagate local event notifications to the remote peer.	efm-oam port propagate-events	<i>critical event</i> - enabled <i>dying-gasp event</i> - enabled.
The threshold, window frame values and notify status for errored frame period events.	efm-oam errored-frame-period	<i>threshold_symbols</i> - 1 frame error <i>window_frames</i> - Depends on port types. <i>notify status</i> - enable
The threshold, window, and notify status for errored frame events.	efm-oam errored-frame	<i>threshold_symbols</i> - 1 frame error <i>window_seconds</i> - 1 second <i>notify status</i> - enable
The threshold, window and notify status for errored-frame-seconds-summary on a port.	efm-oam errored-frame-seconds-summary	<i>threshold_symbols</i> - 1 errored frame second <i>window_seconds</i> - 60 seconds. <i>notify status</i> - enable
The number of frames sent by the current LINK OAM port to the MAC address of the remote port, the delay between the frames sent, and whether or not to start the ping operation.	efm-oam port ll-ping	<i>number</i> - 5 frames <i>milliseconds</i> - 1000

Quick Steps for Configuring LINK OAM

The following steps provide a quick tutorial on how to configure LINK OAM. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

- 1 Enable LINK OAM globally on the switch by using the **efm-oam** command. For example:

```
-> efm-oam enable
```

- 2 Enable LINK OAM protocol for a specific port using the **efm-oam port status** command. For example

```
-> efm-oam port 1/1 status enable
```

- 3 Configure the LINK AOM port to active mode by using the **efm-oam port mode** command. For example:

```
-> efm-oam port 1/1 mode active
```

Note. The above step is optional. By default, LINK OAM mode is active on all ports.

- 4 Configure the timeout interval (keep-alive)for the dynamically learned neighboring devices on the port by using the **efm-oam port keepalive-interval** command. For example:

```
-> efm-oam port 1/1 keepalive-interval 10
```

- 5 Configure the time interval by which the information OAMPDUs should be transmitted out of an LINK OAM enabled port by using the **efm-oam port hello-interval** command. For example:

```
-> efm-oam port 1/1 hello-interval 5
```

- 6 Activate remote loop back processing on the port by using the **efm-oam port remote-loopback** command. For example:

```
-> efm-oam port 1/1 remote-loopback process
```

- 7 Activate propagation of critical events and dying gasp events on the port by using the **efm-oam port propagate-events** command. For example:

```
-> efm-oam port 1/1 propagate-events critical-event enable
```

```
-> efm-oam port 1/1 propagate-events dying-gasp enable
```

Note. The above step is optional. By default, propagation of critical events and dying gasp is enabled on the port.

- 8 Configure the threshold, window frame values and notify status for errored frame period events on the port by using the **efm-oam errored-frame-period** command. For example:

```
-> efm-oam port 1/1 errored-frame-period window 3000000 threshold 1 notify enable
```

- 9 Configure the threshold, window, and notify status for errored frame events on the port by using the **efm-oam errored-frame** command. For example:

```
-> efm-oam port 1/1 errored-frame window 32 threshold 10 notify enable
```

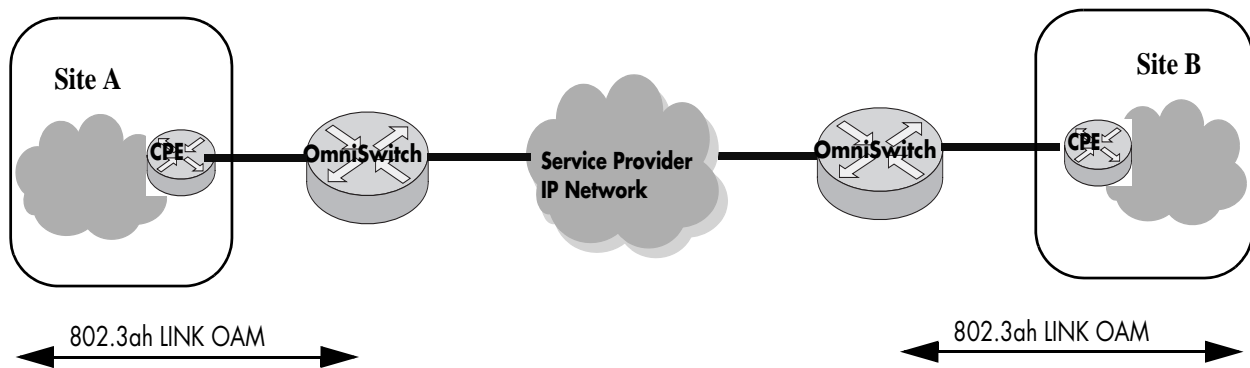

10 Configure the threshold, window and notify-status for errored-frame-seconds-summary on the port by using the **efm-oam errored-frame-seconds-summary** command. For example:

```
-> efm-oam port 1/1 errored-frame-seconds-summary window 700 threshold 1 notify  
enable
```

LINK OAM Overview

IEEE standard 802.3ah provides support for LINK OAM. The Clause 57 of std. 802.3ah defines the Operations, Administration, and Maintenance (OAM) sub layer, which provides mechanisms useful for monitoring link operation such as remote fault indication and remote loopback control. LINK OAM provides network operators the ability to monitor the health of the network and quickly determine the location of failing links or fault conditions.

LINK OAM provides an OAMPDU-based mechanism to notify the remote DTE when one direction of a link is non-operational and therefore data transmission is disabled. The ability to operate a link in a unidirectional mode for diagnostic purposes supports the maintenance objective of failure detection and notification.



Example LINK OAM

OAM information is conveyed in slow protocol frames called OAM Protocol Data Units (OAMPDUs). OAMPDUs contain the appropriate control and status information used to monitor, test and troubleshoot OAM-enabled links. OAMPDUs traverse a single link, being passed between peer OAM nodes, and as such, are not forwarded by MAC clients (e.g., bridges or switches). OAM does not include functions such as station management, bandwidth allocation or provisioning functions.

The mandatory LINK OAM functions include discovery operations (determining if the other end of the link is OAM capable and what OAM functions it supports), state machine implementation and some critical event flows. OAM remote loopback can be used for fault localization and link performance testing.

The features of the LINK OAM protocol discussed in this section are:

- [“Discovery” on page 17-6](#)
- [“Link Monitoring” on page 17-7](#)
- [“Remote Fault detection” on page 17-7](#)
- [“Remote Loopback Testing” on page 17-8](#)

Discovery

Discovery is the first phase of the IEEE 802.3ah OAM protocol. During discovery, information about LINK OAM node’s capabilities, configuration, and identity are exchanged in the form of OAM protocol data units (OAMPDUs).

The interconnected LINK OAM nodes notify the peer of their OAM configuration information and the OAM capabilities of the local nodes by exchanging Information OAMPDUs and determine whether LINK OAM connections can be established. A LINK OAM connection between two nodes is established only when the settings concerning Loopback, link detecting, and link event of the both sides match.

Note. LINK OAM requires that frames be exchanged with a minimum frequency to maintain the relationship(keep-alive). If no OAMPDUs are received in a 5 second window, the OAM peering relationship is lost and must be restored to perform OAM functions. Use **efm-oam port keepalive-interval** command to configure the keepalive time interval.

Link Monitoring

Error detection in an Ethernet network is difficult, especially when the physical connection in the network is not disconnected but network performance is degrading gradually. Link monitoring is used to detect and indicate link faults in various environments. Link monitoring uses the Event Notification OAMPDU, and sends events to the remote OAM node when there is a disorder detected on the link. The error events defined are:

Errored frame event - An errored frame event occurs when the number of detected error frames over a specific interval exceeds the predefined threshold.

Errored frame period event - An errored frame period event occurs if the number of frame errors in specific number of received frames exceeds the predefined threshold.

Errored frame seconds event - When the number of error frame seconds detected on a port over a detection interval reaches the error threshold, an errored frame seconds event occurs.

For configuring errored frame, errored frame period, and errored frame seconds events on a port, see [“Configuring Link Monitoring” on page 17-10](#)

Remote Fault detection

In a network where traffic is interrupted due to device failures or unavailability, the flag field defined in OAMPDUs allows a LINK OAM enabled node to send severe error conditions to its peer. The severe error conditions that can be identified are:

Dying Gasp - This flag is raised when a node is about to reset, reboot, or otherwise go to an operationally down state. (An unexpected fault, such as power failure has occurred.)

Critical Event - This flag indicates a severe error condition that does not result in a complete reset or reboot by the peer node. (An undetermined critical event happened.)

One of the most critical problems in an access network for carriers is differentiating between a simple power failure at the customer premise and an equipment or facility failure. Dying gasp provides this information by having a node indicate to the network that it is having a power failure. More details on the failure may be included in additional event information conveyed in the frame.

For setting up the notification of critical events on a port, see [“Enabling and Disabling Propagation of Events” on page 17-9](#)

Remote Loopback Testing

Remote loopback, which is often used to troubleshoot networks, allows one node to put the other node into a state whereby all inbound traffic is immediately reflected back onto the link. Remote loopback is most useful as a diagnostic tool, where it can be used to isolate problem segments in a large network.

By performing remote loopback tests periodically, network administrators can detect network faults in time and also isolate the network segments where errors have occurred.

Remote loopback testing in networks can be done only after the LINK OAM connection is established. With remote loopback enabled, the LINK OAM node operating in active LINK OAM mode issues remote loopback requests and the peer responds to them. If the peer operates in the loopback mode, it returns all the PDUs except Ethernet OAMPDUs to the senders along the original paths.

For enabling or disabling remote loopback process on a port, see [“Enabling and Disabling Remote loopback” on page 17-11](#)

Interaction With Other Features

This section contains important information about how other OmniSwitch features interact with LINK OAM. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

Link Aggregate

LINK OAM will not work on the logical link aggregate port. But, it can run on the individual aggregable (physical) port.

Connectivity Fault Management

Connectivity Fault Management (IEEE 802.1ag) covers the scope of Ethernet service over any path, whether a single link or end-to-end, enabling service providers to fully monitor Ethernet service regardless of the layers supporting the service, the network path, or the various network operators involved. It divides a network into maintenance domains in the form of hierarchy levels, which are then allocated to users, service providers and operators.

Connectivity Fault Management (CFM) assigns maintenance end points (MEPs) to the edges of each domain and maintenance intermediate points (MIPs) to ports within domains. This helps to define the relationships between all entities from a maintenance perspective, to allow each entity to monitor the layers under its responsibility and localize the errors easily.

ERP

LINK OAM is supported in Ethernet Ring Protection (ERP) switching mechanism. ERP (ITU-T G.8032/Y.1344) is a self-configuring algorithm that maintains a loop-free topology while providing data path redundancy and network scalability. ERP provides fast recovery times for Ethernet ring topologies by utilizing traditional Ethernet MAC and bridge functions..

Configuring LINK OAM

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to configure LINK OAM on a switch.

Enabling and Disabling LINK OAM

The **efm-oam** should be used to enable LINK OAM globally. By default, LINK OAM is disabled on the switch. The **efm-oam port status** command can be used to enable or disable the LINK OAM on a specific port or a range of ports on a switch. When enabled, the port can be set to receive, transmit, or both transmit and receive OAMPDUs.

To enable LINK OAM globally on a range of ports, use the **efm-oam** command, as shown:

```
-> efm-oam port 2/1-10 status enable
```

To disable LINK OAM globally on a range of ports, use the **disable** form of the command, as shown:

```
-> efm-oam port 2/1-10 status disable
```

To enable LINK OAM mode to active, use the **port mode** command, as shown:

```
-> efm-oam port 2/1-10 mode active
```

By default, LINK OAM port mode is active on all the ports.

Setting the Transmit Delay

LINK OAM requires that frames be exchanged with a minimum frequency to maintain the relationship (keep-alive). If no OAMPDUs are received in a specific time interval window, the OAM peering relationship is lost and must be restored to perform OAM functions.

Use **efm-oam port keepalive-interval** command to configure the keepalive time interval.

```
-> efm-oam port 2/1-10 keepalive-interval 10
```

To configure the time interval by which the information OAMPDUs should be transmitted out of an LINK OAM enabled port, use the **efm-oam port hello-interval** command.

```
-> efm-oam port 2/1-10 hello-interval 10
```

Note. By default, the keep-alive interval value is 5 seconds and the hello-interval value is set to 1 second.

Enabling and Disabling Propagation of Events

In a network where traffic is interrupted due to device failures or unavailability, the flag field defined in OAMPDUs allows a LINK OAM enabled node to send severe error conditions to its peer. See [“Remote Fault detection” on page 17-7](#) for more information on error conditions.

The ports can be enabled to report severe error conditions like critical events and dying gasp events by using the **efm-oam port propagate-events** command.

```
-> efm-oam port 2/1-10 propagate-events critical-event enable
```

```
-> efm-oam port 2/1-10 propagate-events dying-gasp enable
```

Note. The above commands are optional. By default, propagation of critical events and dying gasp is enabled on the port.

Configuring Link Monitoring

Link monitoring is used to detect and indicate link faults in various environments. Link monitoring uses the Event Notification OAMPDU, and sends events to the remote OAM node when there is a disorder detected on the link. For more information on error events, see [“Link Monitoring” on page 17-7](#)

Enabling and Disabling Errored frame period

Configure the threshold, window frame values and notify status for errored frame period events on the port by using the [efm-oam errored-frame-period](#) command.

```
-> efm-oam port 2/1-10 errored-frame-period window 3000000 threshold 1 notify enable
```

To disable notification of errored frame period events, use the following command.

```
-> efm-oam port 2/1-10 errored-frame-period notify disable
```

Enabling and Disabling Errored frame

Configure the threshold, window, and notify status for errored frame events on the port by using the [efm-oam errored-frame](#) command.

```
-> efm-oam port 2/1-10 errored-frame window 32 threshold 10 notify enable
```

To disable notification of errored frame events, use the following command.

```
-> efm-oam port 2/1-10 errored-frame notify disable
```

Enabling and Disabling Errored frame seconds summary

Configure the threshold, window and notify-status for errored-frame-seconds-summary on the port by using the [efm-oam errored-frame-seconds-summary](#) command.

```
-> efm-oam port 2/1-10 errored-frame-seconds-summary window 700 threshold 1 notify enable
```

To disable notification of errored frame events, use the following command.

```
-> efm-oam port 2/1-10 errored-frame-seconds-summary notify disable
```

Configuring LINK OAM Loopback

Remote loopback is most useful as a diagnostic tool, where it can be used to isolate problem segments in a large network. See “[Remote Loopback Testing](#)” on page 17-8 for more information.

Enabling and Disabling Remote loopback

LINK OAM loopback testing can be performed only after the LINK OAM connection is established and the hosts are operating in active LINK OAM mode.

When the remote-loopback is in **process** mode, the session started by peer LINK OAM client will be processed by local LINK OAM port. As a result, remote port will be in remote-loopback state and the local port will be local-loopback state.

Activate remote loop back processing on the port by using the **remote-loopback** command.

```
-> efm-oam port 2/1-10 remote-loopback process
```

When the remote-loopback is in **ignore** mode, the session started by peer LINK OAM will not be processed by the local port.

For remote loop back processing to be ignored on the port, use the following command.

```
-> efm-oam port 2/1-10 remote-loopback ignore
```

After configuring the port to process remote loopback, the port should be initiated for loopback session to start.

```
-> efm-oam port 1/1 remote-loopback start
```

The above command will initiate the loopback control PDU towards the peer port to start. To stop the remote-loopback session, use the following command.

```
-> efm-oam port 1/1 remote-loopback stop
```

To configure the number of frames to be sent by the current LINK OAM port to the remote port's MAC address (11 ping) and the delay between each consecutive sent frames and to start the ping operation, use the following command.

```
-> efm-oam port 1/20 11-ping num-frames 12 delay 500 start
```

Note. By default, the number of frames value is 5 frames and the delay is set to 1000 milliseconds.

Verifying the LINK OAM Configuration

To display information about LINK OAM on the switch, use the show commands listed below:

show efm-oam configuration	Displays the global LINK OAM configuration.
show efm-oam port	Displays the status of LINK OAM on all the ports in the system, along with other relevant information such as OAM mode, operational status and loopback status of the port.
show efm-oam port detail	Displays the LINK OAM configuration and other related parameters for a port.
show efm-oam port statistics	Displays the LINK OAM statistics on a port, or a range of ports or on all ports.
show efm-oam port remote detail	Displays the LINK OAM configuration and details of the related parameters of the remote port.
show efm-oam port history	Displays the log of events that have occurred on a port. Use this command to display specific event logs on a port.
show efm-oam port ll-ping detail	Displays the frames lost during a loopback session.

18 Configuring UDLD

UniDirectional Link Detection (UDLD) is a protocol for detecting and disabling unidirectional Ethernet fiber or copper links caused by mis-wiring of fiber strands, interface malfunctions, media converter faults, etc. The UDLD operates at Layer 2 in conjunction with IEEE 802.3's existing Layer 1 fault detection mechanisms.

UDLD is a lightweight protocol that can be used to detect and disable one-way connections before they create dangerous situations such as Spanning Tree loops or other protocol malfunctions. The protocol is mainly used to advertise the identities of all the UDLD-capable devices attached to the same LAN segment and to collect the information received on the ports of each device to determine whether the Layer 2 communication is functioning properly. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, the protocol administratively shuts down the affected port and generates a trap to alert the user.

In This Chapter

This chapter describes how to configure UDLD parameters through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include the following:

- Configuring UDLD on [page 18-6](#).
- Configuring the operational mode on [page 18-7](#).
- Configuring the probe-message advertisement timer on [page 18-7](#).
- Configuring the echo-based detection timer on [page 18-7](#).
- Clearing UDLD statistics on [page 18-8](#).
- Recovering a port from UDLD shutdown on [page 18-8](#).
- Displaying UDLD information on [page 18-8](#).

UDLD Specifications

RFCs supported	Not applicable at this time
IEEE Standards supported	Not applicable at this time
Platforms Supported	OmniSwitch 6250-Metro Models
Probe-message advertisement timer	7 to 90 in seconds
Echo-based detection timer	4 to 15 in seconds
Maximum neighbors per UDLD port	32
Maximum number of UDLD ports per system	128

UDLD Defaults

Parameter Description	Command	Default
UDLD administrative state	udld	Disabled
UDLD status of a port	udld port	Disabled
UDLD operational mode	udld mode	Normal
Probe-message advertisement timer	udld probe-timer	15 seconds
Echo-based detection timer	udld echo-wait-timer	8 seconds

Quick Steps for Configuring UDLD

- 1 To enable the UDLD protocol on a switch, use the **udld** command. For example:

```
-> udld enable
```

- 2 To enable the UDLD protocol on a port, use the **udld port** command by entering **udld port**, followed by the slot and port number, and **enable**. For example:

```
-> udld port 1/6 enable
```

- 3 Configure the operational mode of UDLD by entering **udld port**, followed by the slot and port number, **mode**, and the operational mode. For example:

```
-> udld port 1/6 mode aggressive
```

- 4 Configure the probe-message advertisement timer on port 6 of slot 1 as 17 seconds using the following command:

```
-> udld port 1/6 probe-timer 17
```

Note. *Optional.* Verify the UDLD global configuration by entering the **show udld configuration** command or verify the UDLD configuration on a port by entering the **show udld configuration port** command. For example:

```
-> show udld configuration
Global UDLD Status : Disabled

-> show udld configuration port 1/6
Global UDLD Status: enabled
Port UDLD Status: enabled
Port UDLD State: bidirectional
UDLD Op-Mode: normal
Probe Timer (Sec): 20,
Echo-Wait Timer (Sec): 10
```

To verify the UDLD statistics of a port, use the **show udld statistics port** command. For example:

```
-> show udld statistics port 1/42
UDLD Port Statistics
Hello Packet Send      :8,
Echo Packet Send       :8,
Flush Packet Recvd     :0
UDLD Neighbor Statistics
Neighbor ID    Hello Pkts Recv    Echo Pkts Recv
-----+-----+-----
      1             8             15
      2             8             15
      3             8             21
      4             8             14
      5             8             15
      6             8             20
```

UDLD Overview

UDLD is a Layer 2 protocol used to examine the physical configuration connected through fiber-optic or twisted-pair Ethernet cables. UDLD detects and administratively shuts down the affected port, and alerts the user when a unidirectional link exists. Unidirectional links can create hazardous situations such as Spanning-Tree topology loops caused, for instance, by unwiring of fiber strands, interface malfunctions, media converter's faults, etc.

The UDLD feature is supported on the following port types:

- Copper ports
- Fiber ports

UDLD Operational Mode

UDLD supports two modes of operation: normal and aggressive modes. UDLD works with the Layer 1 mechanisms to determine the physical status of a link. A unidirectional link occurs whenever the traffic sent by a local device is received by its neighbor; but the traffic from the neighbor is not received by the local device.

Normal Mode

In this mode, the protocol depends on explicit information instead of implicit information. If the protocol is unable to retrieve any explicit information, the port is not put in the shutdown state; instead, it is marked as Undetermined. The port is put in the shutdown state only when it is explicitly determined that the link is defective when it is determined on the basis of UDLD-PDU processing that link has become unidirectional. In any such state transition, a trap is raised.

Aggressive Mode

In this mode, UDLD checks whether the connections are correct and the traffic is flowing bidirectionally between the respective neighbors. The loss of communication with the neighbor is considered an event to put the port in shutdown state. Thus, if the UDLD PDUs are not received before the expiry of a timer, the port is put in the UDLD-shutdown state. Since the lack of information is not always due to a defective link, this mode is optional and is recommended only for point-to-point links.

UDLD shuts down the affected interface when one of these problems occurs:

- On fiber-optic or twisted-pair links, one of the interfaces cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the interfaces is down while the other is up.
- One of the fiber strands in the cable is disconnected.

Mechanisms to Detect Unidirectional Links

The UDLD protocol is implemented to correct certain assumptions made by other protocols, and to help the Spanning Tree Protocol to function properly to avoid the creation of dangerous Layer 2 loops.

UDLD uses two basic mechanisms:

- It advertises the identity of a port and learns about its neighbors. This information about the neighbors is maintained in a cache table.
- It sends continuous echo messages in certain circumstances that require fast notifications or fast re-synchronization of the cached information.

Neighbor database maintenance

UDLD learns about other UDLD neighbors by periodically sending a Hello packet (also called an advertisement or probe) on every active interface to inform each device about its neighbors.

When the switch receives a Hello message, the switch caches the information until the age time expires. If the switch receives a new Hello message before the aging of an older cache entry, the switch replaces the older entry with the new one.

Whenever an interface is disabled and UDLD is running, or UDLD is disabled on an interface, or the switch is reset, UDLD clears all the existing cache entries for the interfaces that are affected by the configuration change. UDLD sends a message to the neighbors to flush the part of their caches affected by the status change. The message is intended to synchronize the caches.

Echo detection

UDLD depends on an echo-detection mechanism. UDLD restarts the detection window on its side of the connection and sends echo messages in response to the request, whenever a UDLD device learns about a new neighbor or receives a re-synchronization request from an out-of-sync neighbor. This behavior is the same on all UDLD neighbors because the sender of the echoes expects to receive an echo as a response.

If the detection window ends and no valid response is received, the link will be shut down, depending on the UDLD mode. When UDLD is in normal mode, the link is considered to be undetermined and will not be shut down. When UDLD is in aggressive mode, the link is considered to be unidirectional, and the interface is shut down.

In normal mode, if UDLD is in the advertisement or in the detection phase and all the neighbor cache entries are aged out, UDLD restarts the link-up sequence to re-synchronize with potentially out-of-sync neighbors.

In aggressive mode, if UDLD is in the advertisement or in the detection phase and all the neighbors of a port are aged out, UDLD restarts the link-up sequence to re-synchronize with potentially out-of-sync neighbors. UDLD shuts down the port, after the continuous messages, if the link state is undetermined.

Configuring UDLD

This section describes how to use Command Line Interface (CLI) commands for enabling and disabling UDLD on a switch or port (see “[Enabling and Disabling UDLD](#)” on page 18-6), configuring the operational mode (see “[Configuring mode](#)” on page 18-7), configuring and resetting probe-message advertisement timer (see “[Configuring probe-timer](#)” on page 18-7), configuring and resetting echo-based detection timer (see “[Configuring echo-wait-timer](#)” on page 18-7), clearing the UDLD statistics on a switch or port (see “[Clearing UDLD Statistics](#)” on page 18-8), and recovering a port from UDLD shutdown (see “[Recovering a port from UDLD shutdown](#)” on page 18-8).

Note. See the “UDLD Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of UDLD CLI commands.

Enabling and Disabling UDLD

The following subsections describe how to enable and disable UDLD on a switch or a port.

Enabling UDLD on a switch

By default, UDLD is disabled on all switch ports. To enable UDLD on a switch, use the **udld** command. For example, the following command enables UDLD on a switch:

```
-> udld enable
```

Disabling UDLD on a switch

To disable UDLD on a switch, use the **udld** command with the **disable** parameter. For example, the following command disables UDLD on a switch:

```
-> udld disable
```

Enabling UDLD on a port

By default, UDLD is disabled on all switch ports. To enable UDLD on a port, use the **udld port** command. For example, the following command enables UDLD on port 3 of slot 1:

```
-> udld port 1/3 enable
```

To enable UDLD on multiple ports, specify a range of ports. For example:

```
-> udld port 1/6-10 enable
```

Disabling UDLD on a port

To disable UDLD on a port, use the **udld port** command with the **disable** parameter. For example, the following command disables UDLD on a range of ports:

```
-> udld port 5/21-24 disable
```

Configuring mode

To configure the operational mode, use the **udld mode** command as shown:

```
-> udld mode aggressive
```

For example, to configure the mode for port 4 on slot 2, enter:

```
-> udld port 2/4 mode aggressive
```

To configure the mode for multiple ports, specify a range of ports. For example:

```
-> udld port 2/7-18 mode normal
```

Note. The Normal mode is the default operational mode of UDLD.

Configuring probe-timer

To configure the probe-message advertisement timer, use the **udld probe-timer** command as shown:

```
->udld probe-timer 20
```

For example, to configure the probe-timer for port 3 on slot 6, enter:

```
-> udld port 6/3 probe-timer 18
```

To configure the probe-timer for multiple ports, specify a range of ports. For example:

```
-> udld port 1/8-21 probe-timer 18
```

Use the **no** form of this command to reset the timer. For example, the following command resets the timer for port 4 of slot 6:

```
-> no udld port 6/4 probe-timer
```

The following command resets the timer for multiple ports:

```
-> no udld port 1/8-21 probe-timer
```

Note that when a timer is reset, the default value of 15 seconds is set.

Configuring echo-wait-timer

To configure the echo-based detection timer, use the **udld echo-wait-timer** command as shown:

```
-> udld echo-wait-timer 9
```

For example, to configure the echo-wait-timer for port 5 on slot 6, enter:

```
-> udld port 6/5 echo-wait-timer 12
```

To configure the echo-wait-timer for multiple ports, specify a range of ports. For example:

```
-> udld port 1/8-21 echo-wait-timer 9
```

Use the **no** form of this command to reset the timer. For example, the following command resets the timer for port 6 of slot 4:

```
-> no udld port 4/6 echo-wait-timer
```

The following command resets the timer for multiple ports:

```
-> no udld port 1/8-21 echo-wait-timer
```

Note that when a timer is reset, the default value of 8 seconds is set.

Clearing UDLD Statistics

To clear the UDLD statistics, use the **clear udld statistics port** command. For example, to clear the statistics for port 4 on slot 1, enter:

```
-> clear udld statistics port 1/4
```

To clear the UDLD statistics on all the ports, enter:

```
-> clear udld statistics
```

Recovering a port from UDLD shutdown

To bring a port out of the shutdown state, use the **interfaces clear-violation-all** command. For example, to bring port 5 on slot 1 out of the shutdown state, enter:

```
-> interfaces 1/5 clear-violation-all
```

To bring multiple ports out of the shutdown state, enter:

```
->interfaces 5/5-10 clear-violation-all
```

Displaying UDLD Information

To display UDLD configuration and statistics information, use the show commands listed below:

show udld configuration	Displays the global status of UDLD configuration.
show udld configuration port	Displays the configuration information for all UDLD ports or for a particular UDLD port on the switch.
show udld statistics port	Displays the UDLD statistics for a specific port.
show udld neighbor port	Displays the UDLD neighbor ports.
show udld status port	Displays the UDLD status for all ports or for a specific port.

For more information about the resulting display from these commands, see the *OmniSwitch 6250 CLI Reference Guide*. An example of the output for the **show udld configuration port** and **show udld statistics port** commands is also given in “[Quick Steps for Configuring UDLD](#)” on page 18-3.

19 Configuring MAC Retention

MAC Retention allows a system of stackable switches to retain the MAC address of the primary switch for a fixed or indefinite time, even after multiple takeovers. This minimizes the recalculation of protocols, such as Spanning Tree and Link Aggregation. It also minimizes the updation of tables, such as the Address Resolution Protocol (ARP) table for IPv4 routing and the Neighbor Discovery table for IPv6 routing.

Note. MAC Retention is supported on the OmniSwitch 6250 Series.

In This Chapter

This chapter describes the basic components of MAC Address Retention and how to configure them through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of the commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling MAC Retention on [page 19-6](#).
- Detecting a Duplicate MAC Address on [page 19-6](#).
- Configuring MAC Release on [page 19-6](#).

MAC Retention Defaults

The following table lists the defaults for MAC Retention configuration:

Parameter Description	Command	Default
MAC Address Retention status	mac-retention status	disabled
Status of duplicate MAC Address trap	mac-retention dup-mac-trap	disabled

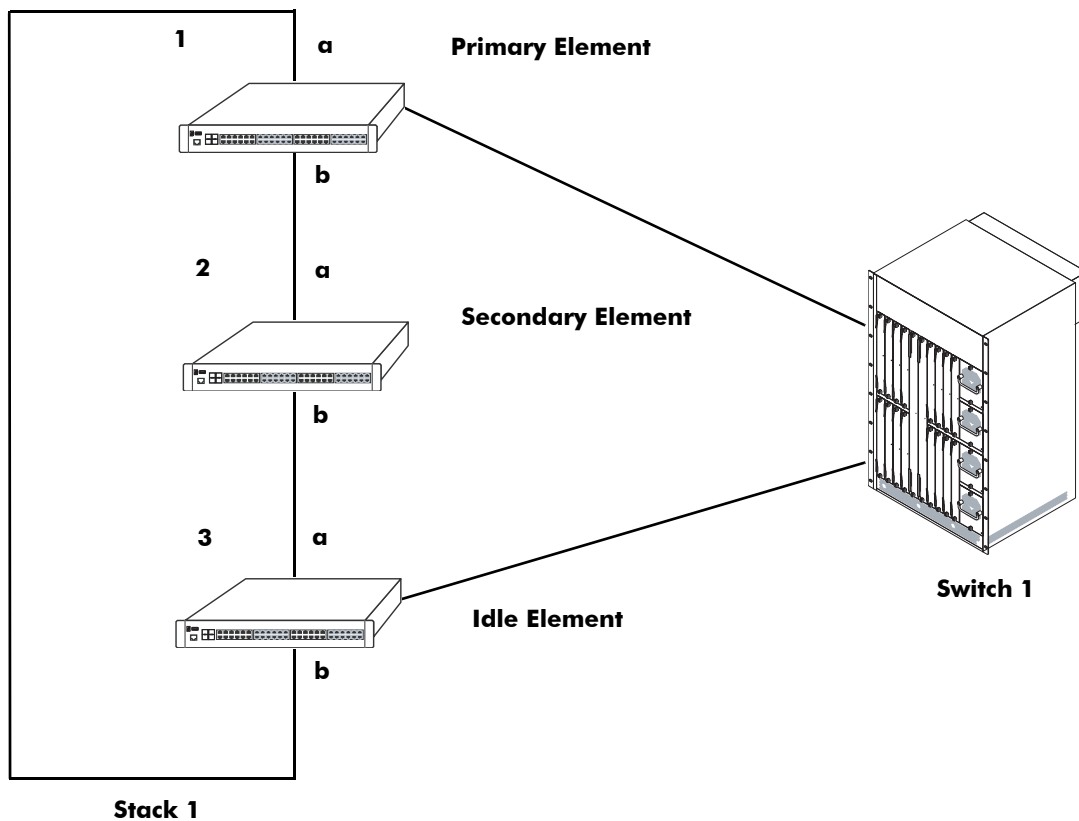
MAC Retention Overview

A “stack element” or simply “element” is a switch that has designated stacking ports. The switches are operatively interconnected via these ports to form a virtual chassis referred to as a *stack*. Each element in a stack can be elected as the primary or the secondary element. The primary element is elected based on the highest uptime or the lowest slot number or the lowest base MAC address. The secondary element is elected based on the lowest slot number or the lowest base MAC address of the remaining elements in the stack. The system of stackable switches is generally coupled in a series and the topology of the system is generally characterized by a closed loop called a ring. A stackable switch is adapted to perform switching between its own data ports and between the data ports of other stackable switches by transmitting packets via the stacking ports.

Each stack element has a unique base MAC address. Generally, the stack address is the MAC address of the current primary element. When a primary element fails, a secondary element starts functioning as the new primary element. This is known as *takeover*. During takeover, the stack address is also accordingly changed to reflect the base MAC address of the new primary element.

Whenever a takeover occurs, it impacts not only the stack, but also the devices that communicate with that stack.

The following diagram shows a stack connected to a stand-alone switch:



Initial State of Stack with 3 Stack Elements

In the above diagram, Stack 1 has the stack address M1. When a takeover occurs, the secondary element starts functioning as the new primary element and the stack address is also changed, for example, to M2, the new primary element’s MAC address. Stack 1 advertises its new stack address M2. Switch 1, which

had previously associated Stack 1 with the stack address M1, now has to change its ARP tables to associate Stack 1 with the new stack address M2.

Similarly, in IPv6 routing, Switch 1 has to change its Neighbor Discovery tables to associate Stack 1 with the new stack address M2.

Another aspect that may be impacted is the recalculation of the Spanning Tree in accordance with the Spanning Tree Protocol (STP). If the stack address is changed due to the election of a new primary element, a new Spanning Tree has to be recalculated to account for this change. This becomes even more difficult when the newly elected primary element becomes the new root bridge.

Link Aggregation Control Protocol (LACP) is another application that is influenced by the takeover. This application uses the base MAC address of the switch as the system ID while exchanging the LACP PDUs in the network. After takeover, the aggregate ports will administratively go down and then come up again due to the change in the system ID.

Therefore, to avoid these recalculations, when a primary element fails in a stack, the secondary element, which takes over as the new primary element uses the MAC address of the former primary element. This feature of retaining the base MAC address of the former primary element for a fixed or indefinite period of time is called MAC Address Retention. In this way, recalculation of protocols, such as Spanning Tree and Link Aggregation and updation of tables, such as the Address Resolution Protocol (ARP) table for IPv4 routing and the Neighbor Discovery table for IPv6 routing is minimized.

Note. The MAC Retention feature is only supported on the switch that operates in the single MAC mode.

How MAC Retention Works

During a full system startup, all the elements in the stack receive the base MAC address read from the EEPROM of the primary element. When the primary element of the stack fails, the secondary element takes over as the new primary element.

This new primary element and all the idle elements of the stack retain this base MAC address. Therefore, this address is called the retained base MAC address.

The ability of the elements to retain this address can be configured, i.e., the MAC Retention feature can be enabled or disabled on the stack. By default, it is disabled.

After a takeover, if the element still uses a retained base MAC address, you can disable the retention process manually. Thereafter, the element will start using the base MAC address from the EEPROM of the currently active primary element.

When the element retains the base MAC address during a takeover, it continues to use this base MAC address irrespective of the return of the former primary element to the stack. This can lead to the duplication of the MAC address.

The duplication of MAC addresses may arise in the following scenarios:

- Failure of non-adjacent elements
- Failure of non-adjacent primary and secondary elements
- Failure of non-adjacent primary and idle elements
- Failure of non-adjacent secondary and idle elements

If the primary element does not return to the stack after the elapse of the specified time interval, a trap is generated, which notifies the administrator of a possible MAC address duplication. The trap and syslog provide details about the slot number and the base MAC address of the removed former primary element.

Note. The duplication of MAC addresses in the network cannot be prevented in case of simultaneous failure of stacking links connected to primary stack element.

MAC Retention After Multiple Take-Overs

After multiple takeovers, if the new primary element still uses the MAC address of the former primary element, you can release the MAC address or disable MAC Retention. In such a case, the stack will obtain a new stack address from the EEPROM of the current primary element.

If you enable the MAC Retention feature again, the old MAC address released earlier will not be retained. Thereafter, the stack will retain the MAC address of the current primary element during future takeovers.

Configuring MAC Retention

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to configure MAC Retention.

Enabling MAC Retention

MAC Retention is disabled on the switch by default. If necessary, use the **mac-retention status** command to enable MAC retention. For example:

```
-> mac-retention status enable
```

To disable MAC Retention on the switch, enter the following:

```
-> mac-retention status disable
```

Note. When the administrative status of MAC retention is enabled, the stack performance is enhanced.

Detecting a Duplicate MAC Address

After a takeover, if the former primary switch does not return to the stack after the preset time interval has elapsed, MAC address duplication may occur. To alert the administrator of a possible MAC address duplication, the switch can be configured to generate an SNMP trap.

You can enable the switch to generate an SNMP trap by using the **mac-retention dup-mac-trap** command as shown:

```
-> mac-retention dup-mac-trap enable
```

To disable SNMP trap generation, enter the following:

```
-> mac-retention dup-mac-trap disable
```

Configuring MAC Release

After multiple takeovers, the switch can be allowed to release the retained MAC address. This enables the stack to obtain a new stack address from the EEPROM of the current primary element.

To release the retained MAC address from a switch, use the **mac release** command as shown:

```
-> mac release
```

Note. A switch will not be allowed to release the MAC address derived from its EEPROM.

To view the MAC Retention status, use the **show mac-retention status** command as shown:

```
-> show mac-retention status
```

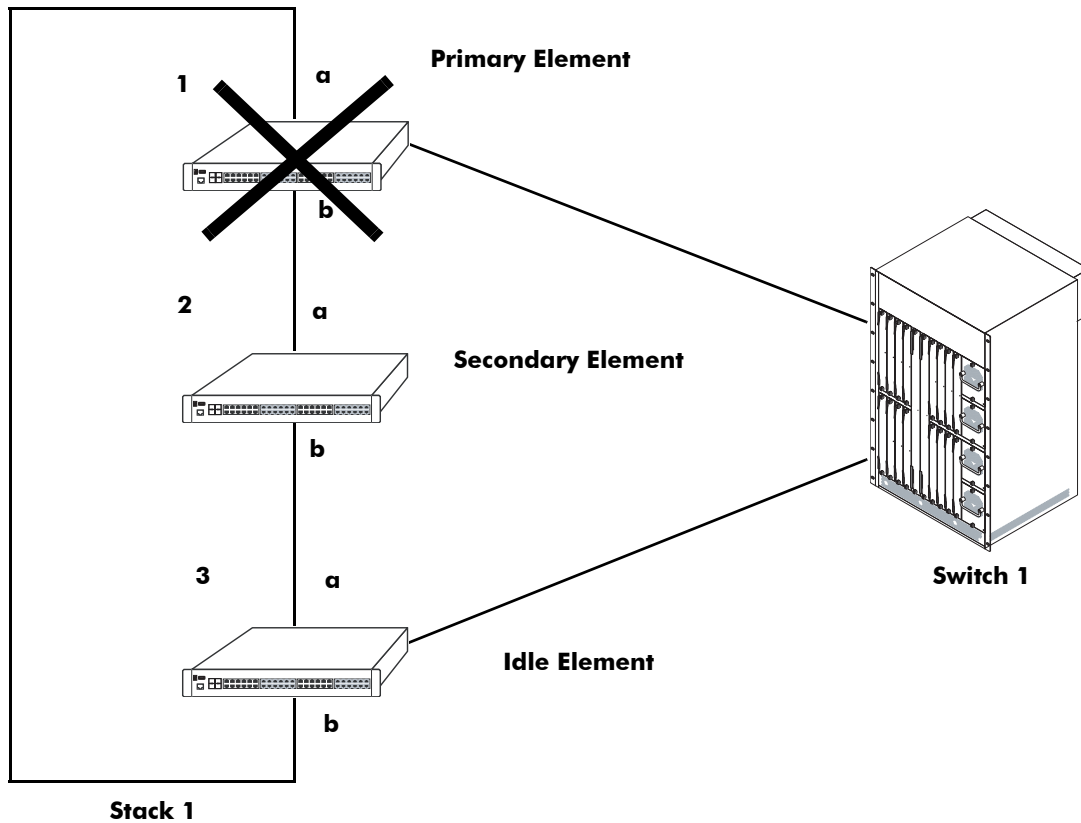
MAC Retention Applications

This section illustrates the MAC Retention feature using two different scenarios:

- **Software Failure**
- **Link Failure**

Software Failure

In the following diagram, if the primary element faces a fatal software exception, the MAC Retention feature will remain enabled and the base MAC address will be retained during takeover.



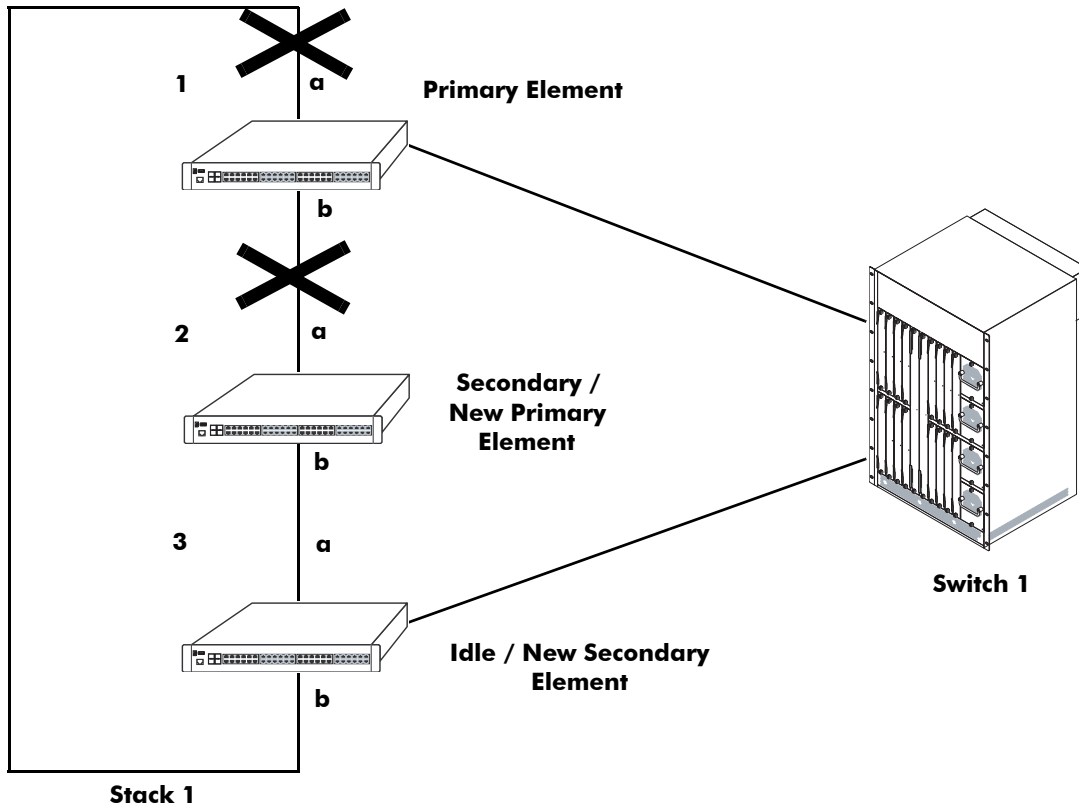
Stack Status when Switch 1 is Down

In the above diagram, when the primary element in Stack 1 fails, the secondary element becomes the new primary element and shares the MAC address of the former primary element of the stack. In this scenario, the decision to retain the base MAC address is acceptable. This feature also works well during the following failures:

- Power failure of the primary element
- Hardware failure of the primary element

Link Failure

In the following diagram, even if both stack links "a" and "b" of the primary element of Stack 1 go down almost at the same time (removed by the user or actual link failures), the MAC Retention feature will remain enabled and the base MAC address will be retained during takeover.



Link Failure

In the above diagram, if the links between the primary and the secondary element and the primary and the idle element fail, the entire stack will split into two separate stacks. The primary element will become an independent stack, and the new primary element (after takeover) and the new secondary element will form another separate stack. Both the stacks will share the same base MAC address. This will lead to the duplication of MAC address because the software running on the elements will not be able to distinguish between a crash or two link failures.

In the above scenario, although the duplication of MAC address cannot be prevented, the element can be configured to generate an SNMP trap. If an SNMP trap is generated, the administrator can release the base MAC address from the stack consisting of the new primary and secondary elements. This stack will use the base MAC address from the EEPROM of the new primary element of the stack.

20 Configuring 802.1AB

Link Layer Discovery Protocol (LLDP) is an emerging standard to provide a solution for the configuration issues caused by expanding networks. LLDP supports the network management software used for complete network management. LLDP is implemented as per the IEEE 802.1AB standard. LLDP specifically defines a standard method for Ethernet network devices to exchange information with its neighboring devices and maintain a database of the information. The exchanged information, passed as LLDPDU, is in TLV (Type, Length, Value) format. The information available to the network management software must be as new as possible; hence, remote device information is periodically updated.

In This Chapter

This chapter describes the basic components of 802.1AB and how to configure them through the Command Line Interface (CLI). The CLI commands are used in the configuration examples; for more details about the syntax of commands, see [Chapter 12, “802.1AB Commands,”](#) in the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include the following:

- [“Quick Steps for Configuring 802.1AB”](#) on page 20-4
- [“Quick Steps for Configuring LLDP-MED Network Policy”](#) on page 20-5
- [“Configuring LLDPDU Flow”](#) on page 20-15.
- [“Nearest Bridge/Edge Mode”](#) on page 20-13
- [“Enabling and Disabling Notification”](#) on page 20-15.
- [“Enabling and Disabling Management TLV”](#) on page 20-16.
- [“Enabling and Disabling 802.1 TLV”](#) on page 20-16.
- [“Enabling and Disabling 802.3 TLV”](#) on page 20-17.
- [“Enabling and Disabling MED TLV”](#) on page 20-17.
- [“Setting the Transmit Interval”](#) on page 20-18.
- [“Setting the Transmit Hold Multiplier Value”](#) on page 20-18.
- [“Setting the Transmit Delay”](#) on page 20-18.
- [“Setting the Transmit Fast Start Count”](#) on page 20-18
- [“Setting the Transmit Fast Start Count”](#) on page 20-18.
- [“Setting the Notification Interval”](#) on page 20-18.
- [“Verifying 802.1AB Configuration”](#) on page 20-19.

802.1AB Specifications

IEEE Specification	<i>IEEE 802.1AB-2005 Station and Media Access Control Connectivity Discovery</i>
Platforms Supported	OmniSwitch 6250 Series
Transmit time interval for LLDPDUs	5 to 32768 in seconds
Transmit hold multiplier value	2 to 10
Transmit delay	1 to 8192 in seconds
Reinit delay	1 to 10 in seconds
Notification interval	5 to 3600 in seconds
Maximum number of network policies that can be associated with a port	8
Maximum number of network policies that can be configured on the switch	32
VLAN ID Range for assigning explicit LLDP-MED Network Policy	1 to 4094
DSCP range	0 to 63
802.1p priority range	0 to 7
Nearest Bridge MAC Address	01:80:c2:00:00:0e
Nearest Edge MAC Address	01:20:da:02:01:73

802.1AB Defaults Table

The following table shows the default settings of the configurable 802.1AB parameters.

Parameter Description	Command	Default Value/Comments
Transmit time interval for LLDPDUs	lldp destination mac-address	30 seconds
Transmit hold multiplier value	lldp transmit hold-multiplier	4
Transmit delay	lldp transmit delay	2 seconds
Transmit Fast Start Count	lldp transmit fast-start-count	3
Reinit delay	lldp reinit delay	2 seconds
Notification interval	lldp notification interval	5 seconds
LLDPDUs transmission	lldp lldpdu	Transmission and Reception
LLDP Network Policy	lldp network-policy	802.1p value: 5 for voice application. 0 for other applications. DSCP value: 0
Per port notification	lldp notification	Disable
Management TLV	lldp tlv management	Disable

Parameter Description	Command	Default Value/Comments
802.1 TLV	lldp tlv dot1	Disable
802.3 TLV	lldp tlv dot3	Disable
LLDP Media Endpoint Device	lldp tlv med	Disable
Mode	lldp destination mac-address	Nearest Bridge

Quick Steps for Configuring 802.1AB

- 1 To enable the transmission and the reception of LLDPUs on a port, use the **lldp lldpdu** command. For example:

```
-> lldp 2/47 lldpdu tx-and-rx
```

- 2 To control per port notification status about the remote device change on a port, use the **lldp notification** command. For example:

```
-> lldp 2/47 notification enable
```

- 3 To control per port management TLV to be incorporated in the LLDPUs, use the **lldp tlv management** command. For example:

```
-> lldp 2/47 tlv management port-description enable
```

- 4 Set the transmit time interval for LLDPUs. To set the timer for a 50 second delay, use the **lldp destination mac-address** command. For example:

```
-> lldp transmit interval 50
```

- 5 Set the minimum time interval between successive LLDPUs. To set the interval for a 20 second delay, use the **lldp transmit delay** command. For example:

```
-> lldp transmit delay 20
```

Note. *Optional.* Verify the LLDP per port statistics by entering the **show lldp statistics** command. For example:

```
-> show lldp statistics
```

Slot/Port	LLDPDU			TLV		Device	
	Tx	Rx	Errors	Discards	Unknown	Discards	Ageouts
1/23	52	0	0	0	0	0	0
2/47	50	50	0	0	0	0	0
2/48	50	50	0	0	0	0	0

To verify the remote system information, use the **show lldp remote-system** command. For example:

```
-> show lldp remote-system
Remote LLDP Agents on Local Slot/Port: 2/47,
Chassis ID Subtype      = 4 (MAC Address),
Chassis ID              = 00:d0:95:e9:c9:2e,
Port ID Subtype         = 7 (Locally assigned),
Port ID                 = 2048,
Port Description        = (null),
System Name             = (null),
System Description      = (null),
Capabilites Supported   = none supported,
Capabilites Enabled     = none enabled,
```

For more information about this display, see the *OmniSwitch CLI Reference Guide*.

Quick Steps for Configuring LLDP-MED Network Policy

Note. A VLAN and VPA must be created for LLDP-MED to work on fixed, mobile or 802.1x ports. However, if the VLAN is not created and the VLAN is added in the LLDP-MED Network Policy, no error is displayed.

LLDP-MED Network Policy for Fixed Ports

Create a VLAN, and associate a port to the VLAN. Subsequently, a network policy ID can be created and associated to the related port. The **lldp tlv med**, **lldp network-policy**, and **lldp med network-policy** commands must be used to configure and enable network policy for fixed ports.

1 Enable the transmission of network policy through a VLAN port using the **lldp tlv med** command. Configure the LLDP-MED TLVs to be transmitted through a particular port using this command. For example:

```
-> lldp 1/10 tlv med network-policy enable
```

2 Configure a local network policy on the switch for a specific application type using the **lldp network-policy** command. Assign a network policy identifier (ID) to a particular application type using this command. For example:

```
-> lldp network-policy 1 application voice vlan 10 12-priority 5
```

3 Bind the network policy to the VLAN port using the **lldp med network-policy** command. For example:

```
-> lldp 1/10 med network-policy 1
```

LLDP on Mobile Ports

For mobile VPA to be created, enable Group Mobility on a port and then define a MAC address rule for an existing VLAN. If the source MAC address of a device matches a MAC address specified in this rule, the device and its mobile port joins the VLAN when the device starts to send traffic.

1 Enable group mobility on a VLAN port using the **vlan** command.

```
-> vlan port mobile 2/10
```

2 Define MAC address rule for the associated VLAN.

```
-> vlan 10 mac mac-address-of-the-lldp-device
```

3 Enable network policy using the **lldp tlv med** command. Configure LLDP-MED TLVs for a particular port using this command.

```
-> lldp 2/10 tlv med network-policy enable
```

4 Configure a local network policy on the switch for a specific application type using the **lldp network-policy** command.

```
-> lldp network-policy 1 application voice vlan 10 12-priority 5
```

- 5 Bind the network policy to a port associated with a VLAN using the **lldp med** command.

```
-> lldp 2/10 med network-policy 1
```

LLDP-MED Network Policy on 802.1x Ports

- 1 Enable group mobility on a VLAN port using the **vlan port** command.

```
-> vlan port mobile 3/10
```

- 2 Enable 802.1x on the VLAN mobile port.

```
-> vlan port 3/10 802.1x enable
```

- 3 Use the **aaa radius-server** command to configure the radius server to be used for port authentication. Configure the radius server to return the VLAN ID for the incoming MAC address of the LLDP device.

```
-> aaa radius-server rad1 host 10.10.2.1 timeout 25
```

- 4 Associate the RADIUS server with authentication for 802.1X ports using the **aaa authentication** command.

```
-> aaa authentication 802.1x rad1
```

- 5 Configure the User Network Profile and add a classification rule for the MAC address using the following command.

```
-> aaa classification-rule mac-address <mac-address-of-the-lldp-device>
user-network-profile name engineering
```

- 6 Enable network policy using the **lldp tlv med** command. Configure LLDP-MED TLVs for a particular port using this command.

```
-> lldp 3/10 tlv med network-policy enable
```

- 7 Configure a local network policy on the switch for a specific application type using the **lldp network policy application** command.

```
-> lldp network-policy 1 application voice vlan 10 l2-priority 5
```

- 8 Bind the network policy to a port associated with a VLAN using the **lldp med** command.

```
-> lldp 3/10 med network-policy 1
```

If the authentication server returns a VLAN ID, then the client device is assigned to the related VLAN.

Note. *Optional.* Verify the LLDP network policies enabled with regard to different network policy IDs, by entering the **show lldp network-policy** command. For example:

```
-> show lldp network-policy
```

```
Legend: 0 Priority Tagged Vlan
        - Untagged Vlan
```

Network Policy ID	Application Type	Vlan Id	Layer2 Priority	DSCP Value
1	voice	10	5	-

802.1AB Overview

LLDP is a Layer 2 protocol for detecting adjacent devices in a network. Each device in a network sends and receives LLDPDUs through all its ports, when the protocol is enabled. If the protocol is disabled on a port or on a device, then LLDPDUs received on that port or device are dropped.

The LLDPDUs are transmitted at a certain interval that can be configured. When an LLDPDU is received from a neighboring device, the LLDPDU software validates the frame and stores the information in its remote device Management Information Base (MIB). This information is aged periodically, if an LLDPDU is not received from the same device within the time mentioned in the TTL TLV of the LLDPDU. By exchanging information with all the neighbors, each device will know its neighbor on each port. The information within the LLDPDU is transmitted in TLV (Type, Length, Value) format and falls under two categories:

- Mandatory
- Optional

Each LLDPDU contains all the four mandatory TLVs and optional TLVs.

Mandatory TLVs

The mandatory TLV's information contains the LAN device's MAC service access point (MSAP) identifier and the time period for the validity of the LAN device's associated information. The mandatory TLVs contained in a LLDPDU are listed below:

- Chassis ID TLV
- Port ID TLV
- VLAN ID TLV
- Time to live TLV
- End of LLDPDU TLV

Optional TLVs

The optional TLVs defined as part of LLDP are grouped into the following sets listed below:

Basic management TLV set

- Port Description TLV
- System Name TLV
- System Description TLV
- System capabilities TLV
- Management address TLV

Note. This optional TLV set is required for all LLDP implementation.

IEEE 802.1 organizationally specific TLV set

- Port VLAN ID TLV
- Port and Protocol VLAN ID TLV
- VLAN name TLV
- Protocol identity TLV

Note. If one TLV from this set is included in the LLDPDU, then all TLVs need to be included.

IEEE 802.3 organizationally specific TLV set

- MAC/PHY configuration/status TLV
- Power Via MDI TLV (In network connectivity TLV set, Extended Power-Via-MDI TLV is supported.)
- Link Aggregation TLV
- Maximum frame size TLV

ANSI-TIA LLDP-MED TLV sets

- Network connectivity TLV set
- LLDP-MED capabilities TLV
- Network Policy TLV
- Location Identification TLV
- Extended Power-via-MDI TLV

When an 802.1AB supporting system receives an LLDPDU containing MED capability TLV, then the remote device is identified as an edge device (IP phone, IP PBX, etc.). In such a case the Alcatel device will stop sending LLDPDU and start sending MED LLDPDU on the port connected to the edge device.

LLDP-Media Endpoint Devices

LLDP-MED is an extension to 802.1ab (Link Layer Discovery Protocol - LLDP), a link-layer protocol that defines a method for network access devices using Ethernet connectivity to advertise device information, device capabilities and media specific configuration information periodically to peer devices attached to the same network.

The LLDP-MED feature facilitates the information sharing between Media Endpoint Devices and Network Infrastructure Devices. It is designed to allow the following functionalities:

- Auto-discovery of LAN policies (such as VLAN, Layer 2 Priority and Diffserv settings) leading to "plug and play" networking. This is achieved by advertising the VLAN information.
- Device location discovery to allow creation of location databases for VoIP, E911 services.
- Extended and automated power management of Power-over-Ethernet endpoints.
- Inventory management, allowing network administrators to track their network devices, and determine their characteristics (manufacturer, software and hardware versions, and serial / asset number).

- Support for receiving, storing and advertising of VLAN information from and to remote Network Connectivity Devices and Media Endpoint Devices (MEDs). LLDP-MED Network Policy TLVs are used to let the OmniSwitch advertise the VLAN to the connected MEDs.
- Support for receiving and storing of Inventory Management TLVs from remote Media Endpoint Devices.

VLAN assignment through explicit LLDP-MED Network Policy is supported on the OmniSwitch AOS.

- The LLDP-MED service advertises the information over the Logical Link-Layer Control Frames and records higher layer management reachability and connection endpoint information from adjacent devices.
- The LLDP-MED service enabled on OmniSwitch operates in advertising mode. However, it does not support any means for soliciting information from the MEDs.

LLDP-MED Network Policy

The network policies for MED devices can be configured on the OmniSwitch using the LLDP-MED CLI commands. A maximum of 32 network policies (0 - 31) can be configured on OmniSwitch. For the feature to work on fixed, mobile and 802.1x ports, there must be a VLAN Port Association (VPA) setup between the VLAN port and the advertised VLAN.

Network Policy - Application Types Supported

Each network policy can be configured with one application type as a mandatory parameter. The following application types are supported:

- Voice
- Voice Signaling
- Guest Voice
- Guest Voice Signaling
- Soft phone voice
- Video Conferencing
- Streaming voice
- Video Signaling

LLDP-MED Network Policy for VLAN Advertisement

The following provisions are provided in the OmniSwitch AOS to assign LLDP-MED network policy for VLAN advertisement:

- The OmniSwitch AOS allows the configuration of a maximum of 32 network policy IDs.
- Each network policy identifier (ID) must be configured with an application type and VLAN-ID as mandatory parameters. Other parameters include L2 priority and DSCP.
- Upto 8 network policy IDs; one per each application type; can be configured for a given port.
- Two or more network policy IDs with the same application type can not be assigned to a port.
- The network policy ID can be configured on fixed, mobile and 802.1x ports.

- When any MED connects to a port with an explicit MED network policy configuration, the OmniSwitch advertises the policy in the LLDPDU along with the MED Network Policy TLVs. This advertisement occurs only if the transmission of the Network Policy TLV is enabled by the user. The Media Endpoint Device must configure itself according to the advertised policy.

Fast Restart of LLDP on Detection of MED

The Fast Restart (as described in IEEE 802.1ab rev) is implemented on the OmniSwitch to transmit the related LLDP-MED Network Policy TLV as soon as a new MED endpoint is detected. The MED TLVs are encapsulated in the LLDPDU. The transmission of LLDP-MED TLV starts only when the OmniSwitch detects a MED capable endpoint on the VLAN port.

LLDP-MED for IP Phones

The LLDP-MED feature on OmniSwitch for voice transmission and VoIP Phones provides a network friendly solution. The information received from and transmitted to IP phones is tagged with voice VLAN ID.

A VLAN can be explicitly assigned to IP Phones through explicit definition of an LLDP-MED network policy identifier. The LLDP-MED Network Policy for the voice and voice signalling application must be activated on the OmniSwitch to advertise the VLAN to the connected IP Phones. For example on how to setup LLDP-MED for IP Phones, see [“Enabling and Disabling Notification” on page 20-15](#)

LLDP Agent Operation

A network device that implements LLDP, supports an LLDP agent. An LLDP agent operates in any one of the following three modes:

Transmit-only mode: The agent can only transmit the information about the capabilities and the current status of the local system at regular intervals.

Receive-only mode: The agent can only receive information about the capabilities and the current status of the remote systems.

Transmit and receive mode: The agent can transmit the capabilities and status information of the local system and receive the capabilities and the status information of the remote system.

LLDPDU Transmission and Reception

LLDP operates in a one-way direction, so that the information in the LLDPDUs flows from one device to another. LLDPDUs are not exchanged as an information request by one device and a response sent by another device. The other devices do not acknowledge LLDP information received from a device.

The transmission of LLDPDU is based on two factors:

- Transmit countdown timing counter. For example, whenever the counter expires, it will go through the entire database of ports that have links and send the LLDPDU if the current time has surpassed the re-transmission time interval.
- If there is change in status of any of the ports. For example, a new port is attached or a new link has come up.

Reception of LLDPDU is a two phase process:

- LLDPDU and TLV error handling as per the 802.1AB standard.

- LLDP remote system MIB update.

Aging Time

The remote system's LLDP specific information is stored in the LLDP MIB. The TTL TLV carries a positive value in seconds, and tells the other device as how long this information is valid. Once a remote device is learned on a local port, if the receiving device doesn't receive an LLDPDU from the same remote device and on the same local port within the TTL mentioned in the previous LLDPDU, then the local device discards that entry from its database. This is called the aging time and can be set by the user.

Nearest Bridge/Edge Mode

Nearest Edge Mode is designed to be used in conjunction with the Automatic Configuration Download feature. By default, when deploying a new switch that does not have any configuration, the Automatic Remote Configuration feature automatically creates a DHCP interface only on the default VLAN. The Nearest Edge mode enhances this functionality and allows the new switch to learn the ID of a management VLAN being advertised by its neighbor and enable the DHCP client interface on a tagged interface for that VLAN.

See the [“Managing Automatic Remote Configuration Download” on page 8-1](#) chapter in the Switch Management Guide for additional information on the Automatic Remote Configuration feature.

The OmniSwitch supports the following two modes:

Nearest-Bridge Mode:

- Nearest-bridge Mode is the default mode for LLDP.
- Nearest-bridge Mode uses the LLDP standard "nearest-bridge" address of 01:80:c2:00:00:0e as the destination MAC address.
- When running in Nearest-bridge Mode LLDP frames with the nearest-edge MAC address are not processed by LLDP but are flooded as normal L2 multicast frames.

Nearest-Edge Mode:

- The switch must be configured to operate in Nearest-edge Mode.
- Nearest-edge Mode uses the Nearest-edge MAC address of 01:20:da:02:01:73 as the destination MAC address, this MAC address is not configurable.
- When LLDP is set to Nearest-edge Mode LLDP frames with a destination mac-address of 01:20:da:02:01:73 are processed by LLDP.
- When running in Nearest-edge Mode LLDP frames with the nearest-bridge MAC address are not processed by LLDP but are flooded as normal L2 multicast frames.

Nearest-Edge Mode Operation

In order for the network to propagate Nearest-edge Mode LLDP PDUs a Management Switch must be configured to send the LLDP PDUs with the management VLAN information. Additionally, the Access Switch is automatically configured to process the Nearest-edge Mode LLDP PDU frames by the Automatic Configuration Download feature.

LLDP Transmisson By The Management Switch

- The Management Switch is configured to use the Nearest-edge Mode MAC address using the **lldp destination mac-address** command and is connected to the network using an untagged interface.
- LLDP is configured on the untagged port of the Management Switch so that the LLDP PDUs are sent with the management VLAN information.
- The LLDP interval should not be set higher than 30 seconds (default).
- The Management Switch sends LLDP PDUs on the untagged interface with the MAC address of 01:20: DA: 02:01:73.

LLDP Propagation By The Network

- These LLDP PDUs are propagated throughout the network as normal L2 multicast frames, eventually reaching the Access Switch.

LLDP Reception By The Access Switch

- The Automatic Configuration Download feature enables the processing of the Nearest-edge LLDP PDUs by default.

See the [“Managing Automatic Remote Configuration Download” on page 8-1](#) chapter in the Switch Management Guide for a configuration example using the Nearest-Edge Mode with the Automatic-Configuration feature.

Configuring 802.1AB

The following sections detail procedures for enabling 802.1AB and assigning ports to 802.1AB.

Configuring LLDPDU Flow

The **lldp lldpdu** command can be used to enable or disable the LLDPDU flow on a specific port, a slot, or all ports on a switch. When enabled, the port can be set to receive, transmit, or both transmit and receive LLDPDUs.

To set the LLDPDU flow on a switch as transmit and receive, enter the **lldp lldpdu** command, as shown:

```
-> lldp chassis lldpdu tx-and-rx
```

To set the LLDPDU flow on port 4 of slot 3 as receive, enter the following command at the CLI prompt:

```
-> lldp 3/4 lldpdu rx
```

To disable the flow of LLDPDU on a switch, enter the **lldp lldpdu** command, as shown:

```
-> lldp chassis lldpdu disable
```

To disable the flow of LLDPDU on port 5 of slot 1, enter the following command at the CLI prompt:

```
-> lldp 1/5 lldpdu disable
```

Enabling and Disabling Notification

The **lldp notification** command is used to control per port notification status about the remote device change on a specific port, a slot, or all ports on a switch. When enabled, the LLDPDU administrative status must be in the receive state.

To enable notification of local system MIB changes on a switch, enter the **lldp notification** command, as shown:

```
-> lldp chassis notification enable
```

To enable notification on port 2 of slot 1, enter the following command at the CLI prompt:

```
-> lldp 1/2 notification enable
```

To disable notification on a switch, enter the **lldp notification** command, as shown:

```
-> lldp chassis notification disable
```

To disable notification on port 4 of slot 1, enter the following command at the CLI prompt:

```
-> lldp 1/4 notificaition disable
```

Enabling and Disabling Management TLV

The **lldp tlv management** command is used to control per port management TLVs transmission in the LLDPDUs on a specific port, a slot, or all ports on a switch. When enabled, the LLDPDU administrative status must be in the transmit state.

To enable the management TLV LLDPDU transmission on a switch, enter the **lldp tlv management** command, as shown:

```
-> lldp chassis tlv management port-description enable
```

To enable the management TLV on port 3 of slot 2, enter the following command at the CLI prompt:

```
-> lldp 2/3 tlv management system-capabilities enable
```

To disable the management TLV on a switch, enter the **lldp tlv management** command, as shown:

```
-> lldp chassis tlv management port-description disable
```

To disable management TLV on port 3 of slot 2, enter the following command at the CLI prompt:

```
-> lldp 2/3 tlv management system-capabilities disable
```

Enabling and Disabling 802.1 TLV

The **lldp tlv dot1** command is used to control per port 802.1 TLVs transmission in the LLDPDUs on a specific port, a slot, or all ports on a switch. When enabled, the LLDPDU administrative status must be in the transmit state.

To enable the 802.1 TLV LLDPDU transmission on a switch, enter the **lldp tlv dot1** command, as shown:

```
-> lldp chassis tlv dot1 port-vlan enable
```

To enable the 802.1 TLV on port 1 of slot 5, enter the following command at the CLI prompt:

```
-> lldp 5/1 tlv dot1 vlan-name enable
```

To disable the 802.1 TLV on a switch, enter the **lldp tlv dot1** command, as shown:

```
-> lldp chassis tlv dot1 port-vlan disable
```

To disable 802.1 TLV on port 2 of slot 5, enter the following command at the CLI prompt:

```
-> lldp 5/2 tlv dot1 vlan-name disable
```


Enabling and Disabling 802.3 TLV

The **lldp tlv dot3** command is used to control per port 802.3 TLVs transmission in the LLDPDUs on a specific port, a slot, or all ports on a switch. When enabled, the LLDPDU administrative status must be in the transmit state.

To enable the 802.3 TLV LLDPDU transmission on a switch, enter the **lldp tlv dot3** command, as shown:

```
-> lldp chassis tlv dot3 mac-phy enable
```

To enable the 802.3 TLV on port 4 of slot 2, enter the following command at the CLI prompt:

```
-> lldp 2/4 tlv dot3 mac-phy enable
```

To disable the 802.3 TLV on a switch, enter the **lldp tlv dot3** command, as shown:

```
-> lldp chassis tlv dot3 mac-phy disable
```

To disable 802.3 TLV on port 5 of slot 3, enter the following command at the CLI prompt:

```
-> lldp 3/5 tlv dot3 mac-phy disable
```

Enabling and Disabling MED TLV

The **lldp tlv med** command is used to control per port LLDP Media End Device (MED) TLVs transmission in the LLDPDUs on a specific port, a slot, or all ports on a switch. When enabled, the LLDPDU administrative status must be in the transmit state.

To enable the LLDP-MED TLV LLDPDU transmission on a switch, enter the **lldp tlv med** command, as shown:

```
-> lldp chassis tlv med power enable
```

To enable the MED TLV on port 4 of slot 4, enter the following command at the CLI prompt:

```
-> lldp 4/4 tlv med capability enable
```

To disable the MED TLV on a switch, enter the **lldp tlv med** command, as shown:

```
-> lldp chassis tlv med power disable
```

To disable MED TLV on port 3 of slot 4, enter the following command at the CLI prompt:

```
-> lldp 4/3 tlv med capability disable
```

To enable the voice application network policy for a MED TLV on the port 3 of slot 4, enter the following command at the CLI prompt:

```
-> lldp 4/3 tlv med network policy 1 enable
```

To disable a MED TLV voice network policy on the port 3 of slot 4, enter the following command at the CLI prompt:

```
-> lldp 4/3 tlv med network policy 1 disable
```

Setting the Transmit Interval

To set the transmit time interval for LLDPDUs, enter the **lldp transmit interval** command. For example, to set the transmit time interval as 40 seconds, enter:

```
-> lldp transmit interval 40
```

Setting the Transmit Hold Multiplier Value

To set the transmit hold multiplier value, enter the **lldp transmit hold-multiplier** command. For example, to set the transmit hold multiplier value to 2, enter:

```
-> lldp transmit hold-multiplier 2
```

Note: The Time To Live is a multiple of the transmit interval and transmit hold-multiplier.

Setting the Transmit Delay

To set the minimum time interval between successive LLDPDUs transmitted, enter the **lldp transmit delay** command. For example, to set the transmit delay value to 20 seconds, enter:

```
-> lldp transmit delay 20
```

By default, the transmit delay is less than or equal to the multiplication of the transmit interval and 0.25.

Setting the Transmit Fast Start Count

To set the fast start count in order to transmit the LLDP-MED Network Policy TLV in LLDPDU as soon as the OmniSwitch detects a new MED capable endpoint device, enter the **lldp transmit fast-start-count** command.

```
-> lldp transmit fast-start-count 3
```

Setting the Reinit Delay

To set the time interval that must elapse before the current status of a port is reinitialized after a status change, enter the **lldp reinit delay** command. For example, to set the reinit delay to 7 seconds, enter:

```
-> lldp reinit delay 7
```

Setting the Notification Interval

To set the time interval that must elapse before a notification about the local system Management Information Base (MIB) change is generated, enter the **lldp notification interval** command. For example, to set the notification value to 130 seconds, enter:

```
-> lldp notification interval 130
```

Note: In a specified interval, generating more than one notification-event is not possible.

Verifying 802.1AB Configuration

To display information about the ports configured to handle 802.1AB, use the following show command:

show lldp config	Displays system-wide statistics.
show lldp statistics	Displays per port statistics.
show lldp local -system	Displays local system information.
show lldp local -port	Displays per port information.
show lldp local-management-address	Displays the local management address information.
show lldp network-policy	Displays the MED Network Policy details for a given policy ID.
show lldp med network-policy	Displays the network policy configured on a slot or port. If no option is specified, network policies configured on all ports of the chassis are displayed.
show lldp remote-system	Displays per local port and information of remote system.
show lldp remote-system med	Displays MED local port information of remote system.

For more information about the resulting display, see [Chapter 12, “802.1AB Commands,”](#) in the *OmniSwitch 6250 CLI Reference Guide*.

21 Using Interswitch Protocols

Alcatel-Lucent Interswitch Protocol (AIP) is used to discover adjacent switches in the network. The following protocol is supported:

- Alcatel-Lucent Mapping Adjacency Protocol (AMAP), which is used to discover the topology of OmniSwitches and Omni Switch/Router (Omni S/R). See [“AMAP Overview” on page 21-3](#).

This protocol is described in detail in this chapter.

In This Chapter

This chapter describes the AMAP protocol and how to configure it through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Activating AMAP on [page 21-5](#).
- Configuring the AMAP discovery time-out interval on [page 21-5](#).
- Configuring the AMAP common time-out interval on [page 21-6](#).

For information about statically and dynamically assigning switch ports to VLANs, see [Chapter 6](#), [“Assigning Ports to VLANs.”](#)

For information about defining VLAN rules that allow dynamic assignment of mobile ports to a VLAN, see [Chapter 8](#), [“Defining VLAN Rules.”](#)

AIP Specifications

Standards	Not applicable at this time. AMAP is an Alcatel-Lucent proprietary protocol.
Platforms Supported	OmniSwitch 6250 Series
Maximum number of IP addresses propagated by AMAP	255

AMAP Defaults

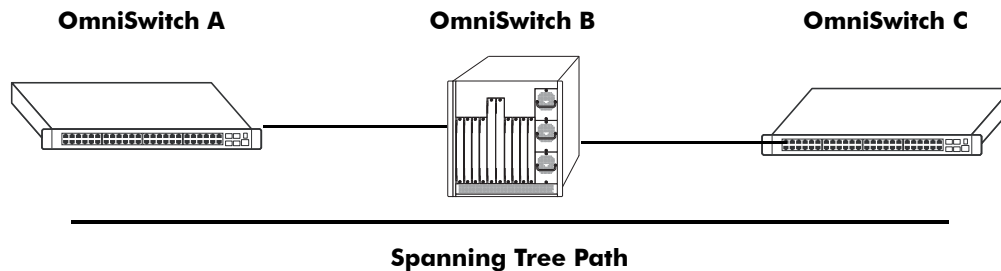
Parameter Description	Command	Default
AMAP status	amap	Enabled
Discovery time interval	amap discovery time	30 seconds
Common time interval	amap common time	300 seconds

AMAP Overview

The Alcatel-Lucent Mapping Adjacency Protocol (AMAP) is used to discover the topology of OmniSwitches in a particular installation. Using this protocol, each switch determines which OmniSwitches are adjacent to it by sending and responding to Hello update packets. For the purposes of AMAP, adjacent switches are those that:

- have a Spanning Tree path between them
- do not have any switch between them on the Spanning Tree path that has AMAP enabled

In the illustration here, all switches are on the Spanning Tree path. OmniSwitch A and OmniSwitch C have AMAP enabled. OmniSwitch B does not. OmniSwitch A is adjacent to OmniSwitch C and vice versa. If OmniSwitch B enables AMAP, the adjacency changes. OmniSwitch A would be next to OmniSwitch B, B would be adjacent to both A and C, and C would be adjacent to B.

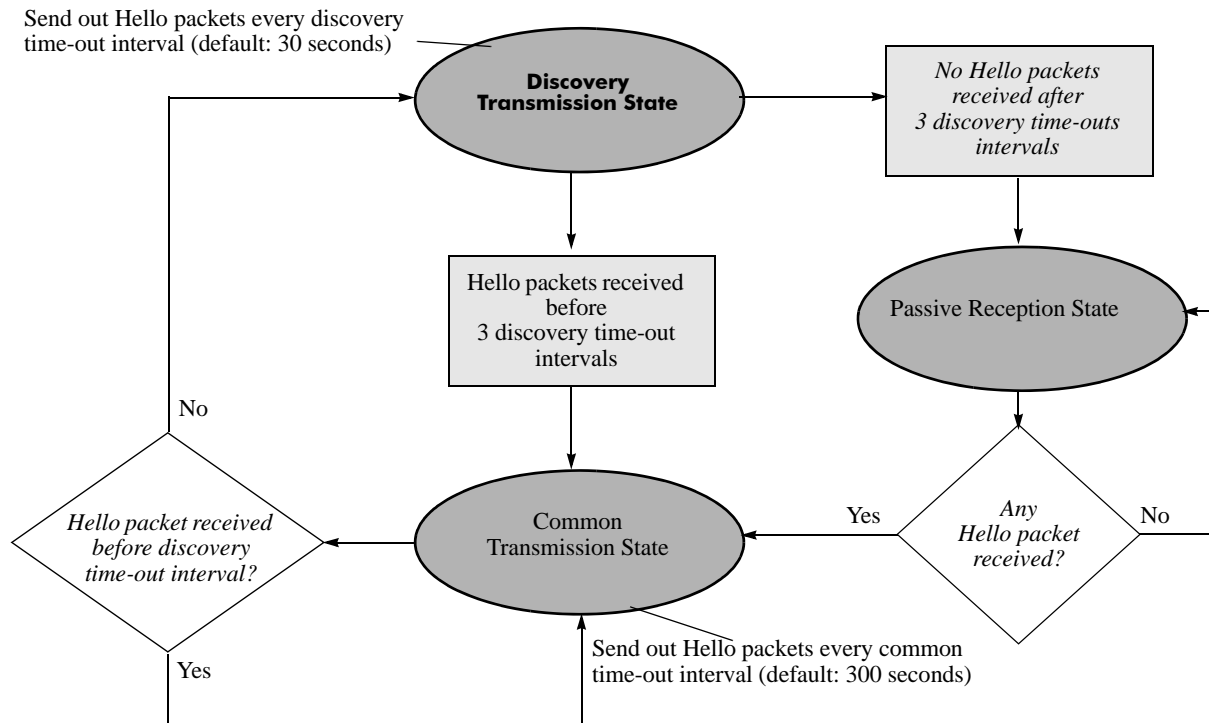


AMAP Transmission States

AMAP switch ports are either in the *discovery transmission state*, *common transmission state*, or *passive reception state*. Ports transition to these states depending on whether or not they receive Hello responses from adjacent switches.

Note. All Hello packet transmissions are sent to a well-known MAC address (0020da:007004).

The transmission states are illustrated on [page 21-3](#).



AMAP Transmission States

Discovery Transmission State

When AMAP is active, at startup all active switch ports are in the discovery transmission state. In this state, ports send out Hello packets and wait for Hello responses. Ports send out Hello packets at a configurable interval called the *discovery time-out interval*. This interval is 30 seconds by default. The ports send out Hello packets up to *three* time-outs of this interval trying to discover adjacent switches.

Any switch ports that receive Hello packets send a Hello response and transition to the common transmission state. Any switch ports that do not receive a Hello response before three discovery time-out intervals have expired are placed in the passive reception state.

Common Transmission State

In the common transmission state, ports detect adjacent switch failures or disconnects by sending Hello packets and waiting for Hello responses. Ports send out Hello packets at a configurable interval called the *common time-out interval*. This interval is 300 seconds by default. To avoid synchronization with adjacent switches, the common time-out interval is jittered randomly by plus or minus ten percent.

Ports wait for a Hello response using the discovery time-out interval. If a Hello response is detected within one discovery time-out interval, the port remains in the common transmission state. If a Hello response is not detected within one discovery time-out interval, the port reverts to the discovery transmission state.

Passive Reception State

In the passive reception state, switch ports are in receive-only mode. Hello packets are not sent out from ports in this state and there is no timer on waiting for Hello responses. If the port receives a Hello packet at any time, it enters the common transmission state and transmits a Hello packet in reply.

If a port transitions to the passive reception state, any remote switch entries for that port are deleted.

Common Transmission and Remote Switches

If an AMAP switch is connected to multiple AMAP switches via a hub, the switch sends and receives Hello traffic to and from the remote switches through the same port. If one of the remote switches stops sending Hello packets and other remote switches continue to send Hello packets, the ports in the common transmission state will remain in the common transmission state.

The inactive switch will eventually be aged out of the switch's AMAP database because each remote switch entry has a "last seen" field that is updated when Hello packets are received. The switch checks the "last seen" field at least once every common time-out interval. Switch ports that are no longer "seen" may still retain an entry for up to three common time-out intervals. The slow aging out prevents the port from sending Hello packets right away to the inactive switch and creating additional unnecessary traffic.

Configuring AMAP

AMAP is active by default. In addition to disabling or enabling AMAP, you can view a list of adjacent switches or configure the time-out intervals for Hello packet transmission and reception.

Enabling or Disabling AMAP

To display whether or not AMAP is active or inactive, enter the following command:

```
-> show amap
```

To activate AMAP on the switch, enter the following command:

```
-> amap enable
```

To deactivate AMAP on the switch, enter the following command:

```
-> amap disable
```

Configuring the AMAP Discovery Time-out Interval

The discovery time-out interval is used in both the discovery transmission state and the common transmission state to determine how long the port will wait for Hello packets. For ports in the discovery transmission state, this timer is also used as the interval between sending out Hello packets.

Note. Ports in the common transmission state send out Hello packets based on the common time-out interval described later.

The discovery time-out interval is set to 30 seconds by default. To display the current discovery time-out interval, enter the following command:

```
-> show amap
```

To change the discovery time-out interval, use either of these forms of the command with the desired value (any value between 1 and 65535). Note that the use of the **time** command keyword is optional. For example:

```
-> amap discovery 60
-> amap discovery time 60
```

Configuring the AMAP Common Time-out Interval

The common time-out interval is used only in the common transmission state to determine the time interval between sending Hello update packets. A switch sends an update for a port just before or after the common time-out interval expires.

Note. Switches avoid synchronization by jittering the common time-out interval plus or minus 10 percent of the configured value. For example, if the default common time-out interval is used (300 seconds), the jitter is plus or minus 30 seconds.

When a Hello packet is received from an adjacent switch before the common time-out interval expires, the switch sends a Hello reply and restarts the common transmission timer.

The common time-out interval is set to 300 seconds by default. To display the current common time-out interval, enter the following command:

```
-> show amap
```

To change the common time-out interval, use either of these forms of the command with the desired value (any value between 1 and 65535). Note that the use of the **time** command keyword is optional. For example:

```
-> amap common 600
-> amap common time 600
```

Displaying AMAP Information

Use the **show amap** command to view a list of adjacent switches and their associated MAC addresses, interfaces, VLANs, and IP addresses. For remote switches that stop sending Hello packets and that are connected via a hub, entries may take up to three times the common time-out intervals to age out of this table.

The following example shows three interfaces on a local AMAP switch (4/1, 5/1, 7/1) connected to interfaces on two remote switches. Interface 5/1 is connected to a remote switch through a hub.

```
-> show amap

AMAP:
  Operational Status = enabled,
  Common Phase Timeout Interval (seconds) = 300,
  Discovery Phase Timeout Interval (seconds) = 30

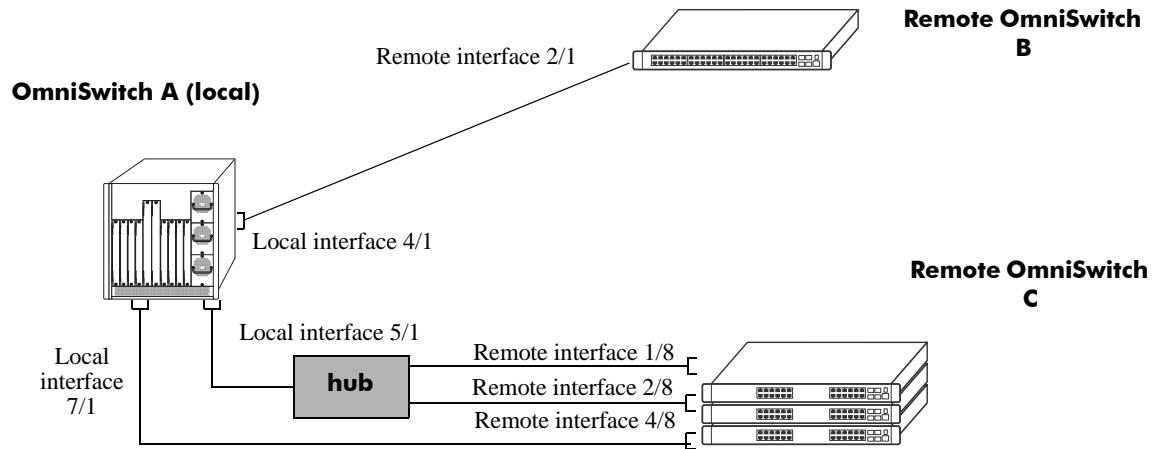
Remote Host 'OmniSwitch B' On Port 4/1 Vlan 1:
Remote Device      = OS6250,
Remote Base MAC    = 00:20:da:03:2c:40,
Remote Interface   = 2/1,
Remote VLAN        = 1,
Number of Remote IP Address(es) Configured = 4,
Remote IP(s) =
18.1.1.1
27.0.0.2
192.168.10.1
192.206.184.40

Remote Host 'OmniSwitch C' On Port 5/1 Vlan 7:
Remote Device      = OS6250,
Remote Base MAC    = 00:20:da:99:96:60,
Remote Interface   = 1/8,
Remote Vlan        = 7,
Number of Remote IP Address(es) Configured = 1,
Remote IP(s) =
192.206.184.20

Remote Host 'OmniSwitch C' On Port 5/1 Vlan 7:
Remote Device      = OS6250,
Remote Base MAC    = 00:20:da:99:96:60,
Remote Interface   = 2/8,
Remote Vlan        = 255,
Number of Remote IP Address(es) Configured = 1,
Remote IP(s) =
192.206.185.30

Remote Host 'OmniSwitch C' On Port 7/1 Vlan 455:
Remote Device      = OS6250,
Remote Base MAC    = 00:20:da:99:96:60,
Remote Interface   = 4/8,
Remote Vlan        = 455,
Number of Remote IP Address(es) Configured = 3,
Remote IP(s) =
192.206.183.10
192.206.184.20
192.206.185.30
```

A visual illustration of these connections is shown here:



AMAP Application Example

See the *OmniSwitch 6250 CLI Reference Guide* for information about the **show amap** command.

22 Configuring 802.1Q

802.1Q is the IEEE standard for segmenting networks into VLANs. 802.1Q segmentation is done by adding a specific tag to a packet.

In this Chapter

This chapter describes the basic components of 802.1Q VLANs and how to configure them through the Command Line Interface (CLI). The CLI commands are used in the configuration examples; for more details about the syntax of commands, see “802.1Q Commands” in the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Setting up an 802.1Q VLAN for a specific port. See [“Enabling Tagging on a Port” on page 22-5](#).
- Setting up an 802.1Q VLAN for a link aggregation group. See [“Enabling Tagging with Link Aggregation” on page 22-5](#).
- Configuring 802.1Q VLAN parameters. See [“Configuring the Frame Type” on page 22-6](#).

For information on creating and managing VLANs, see [Chapter 4, “Configuring VLANs.”](#)

For information on creating and managing link aggregation groups, see [Chapter 23, “Configuring Static Link Aggregation”](#) and [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

802.1Q Specifications

IEEE Specification	<i>Draft Standard P802.1Q/D11 IEEE Standards for Local And Metropolitan Area Network: Virtual Bridged Local Area Networks, July 30, 1998</i>
Platforms Supported	OmniSwitch 6250 Series
Maximum Tagged VLANs per Port	4093
Maximum Untagged VLANs per Port	One untagged VLAN per port.
Maximum VLAN Port Associations (VPA) per switch	32768
Maximum 802.1Q VLAN port associations per switch	2500
Force Tag Internal	Not configurable on the OmniSwitch 6250 Series.

Note. Up to 4093 VLANs can be assigned to a tagged port or link aggregation group. However, each assignment counts as a single VLAN port association. Once the maximum number of VLAN port associations is reached, no more VLANs can be assigned to ports. For more information, see the chapter titled [Chapter 6, “Assigning Ports to VLANs.”](#)

802.1Q Defaults Table

The following table shows the default settings of the configurable 802.1Q parameters.

802.1Q Defaults

Parameter Description	Command	Default Value/Comments
What type of frames accepted	vlan 802.1q frame type	Both tagged and untagged frames are accepted

802.1Q Overview

Alcatel-Lucent's 802.1Q is an IEEE standard for sending frames through the network tagged with VLAN identification. This chapter details procedures for configuring and monitoring 802.1Q tagging on a single port in a switch or a link aggregation group in a switch.

802.1Q tagging is the IEEE version of VLANs. It is a method for segregating areas of a network into distinct VLANs. By attaching a label or tag to a packet, the packet can be identified as being from a specific area or identified as being destined for a specific area.

When enabling a tagged port, you will also need to specify whether only 802.1Q tagged traffic is allowed on the port, or whether the port accepts both tagged and untagged traffic.

“Tagged” refers to four bytes of reserved space in the header of the packet. The four bytes of “tagging” are broken down as follows: the first two bytes indicate whether the packet is an 802.1Q packet, and the next two bytes carry the VLAN identification (VID) and priority.

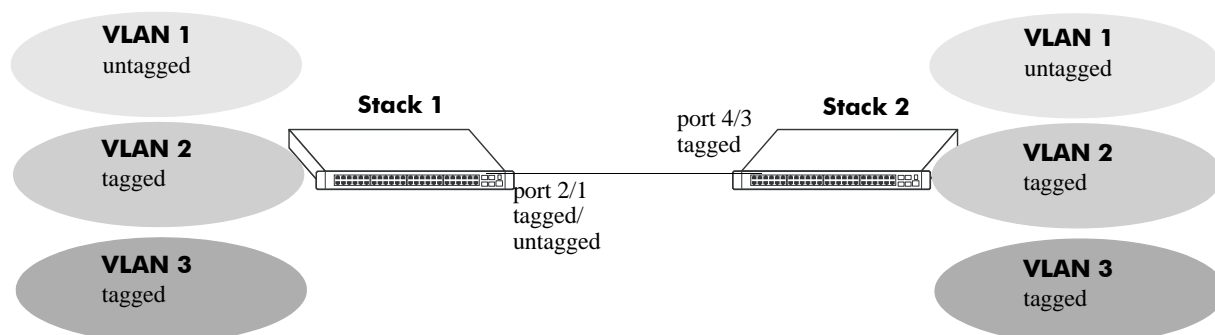
On the ingress side, packets are classified in a VLAN. After classifying a packet, the switch adds an 802.1Q header to the packet. Egress processing of packets is done by the switch hardware. Packets have an 802.1Q tag, which may be stripped off based on 802.1Q tagging/stripping rules.

If a port is configured to be a tagged port, then all the untagged traffic (including priority tagged or VLAN 0 traffic) received on the port will be dropped. You do not need to reboot the switch after changing the configuration parameters.

Note. Priority tagged traffic or traffic from VLAN 0 is used for Quality of Service (QoS) functionality. 802.1Q views priority tagged traffic as untagged traffic.

Mobile ports can be configured to accept 802.1Q traffic by enabling the VLAN mobile tagging feature as described in [Chapter 4, “Configuring VLANs.”](#)

The following diagram illustrates a simple network by using tagged and untagged traffic:



Tagged and Untagged Traffic Network

Stack 1 and 2 have three VLANs, one for untagged traffic and two for tagged traffic. The ports connecting Stack 1 and 2 are configured in such a manner that Port 4/3 will only accept tagged traffic, while Port 2/1 will accept both tagged and untagged traffic.

The port can only be assigned to one untagged VLAN (in every case, this will be the default VLAN). In the example above the default VLAN is VLAN 1. The port can be assigned to as many 802.1Q VLANs as necessary, up to 4093 per port or 32768 VLAN port associations.

For the purposes of Quality of Service (QoS), 802.1Q ports are always considered to be *trusted* ports. For more information on QoS and trusted ports, see [Chapter 34, “Configuring QoS.”](#)

Alcatel-Lucent’s 802.1Q tagging is done at wire speed, providing high-performance throughput of tagged frames. The procedures below use CLI commands that are thoroughly described in “802.1Q Commands” of the *OmniSwitch 6250 CLI Reference Guide*.

Configuring an 802.1Q VLAN

The following sections detail procedures for creating 802.1Q VLANs and assigning ports to 802.1Q VLANs.

Enabling Tagging on a Port

To set a port to be a tagged port, you must specify a VLAN identification (VID) number and a port number. You may also optionally assign a text identification.

For example, to configure port 4 on slot 3 to be a tagged port, enter the following command at the CLI prompt:

```
-> vlan 5 802.1q 3/4
```

Tagging would now be enabled on port 3/4, with a VID of 5.

To add tagging to a port and label it with a text name, you would enter the text identification following the slot and port number. For example, to enable tagging on port 4 of slot 3 with a text name of **port tag**, enter the command in the following manner:

```
-> vlan 5 802.1q 3/4 "port tag"
```

The tagged port would now also be labeled **port tag**. Note that you must use quotes around the text description.

The VLAN used to handle traffic on the tagged port must be created prior to using the **vlan 802.1q** command. Creating a VLAN is described in [Chapter 4, “Configuring VLANs.”](#)

For more specific information, see the **vlan 802.1q** command section in the *OmniSwitch 6250 CLI Reference Guide*.

Enabling Tagging with Link Aggregation

To enable tagging on link aggregation groups, enter the link aggregation group identification number in place of the slot and port number, as shown:

```
-> vlan 5 802.1q 8
```

(For further information on creating link aggregation groups, see [Chapter 23, “Configuring Static Link Aggregation,”](#) or [Chapter 24, “Configuring Dynamic Link Aggregation.”](#))

To add tagging to a port or link aggregation group and label it with a text name enter the text identification following the slot and port number or link aggregation group identification number. For example, to enable tagging on link aggregation group 8 with a text name of **agg port tag**, enter the command in the following manner:

```
-> vlan 5 802.1q 8 "agg port tag"
```

The tagged port would now also be labeled **agg port tag**. Note that you must use quotes around the text description.

To remove 802.1Q tagging from a selected port, use the same command as above with a **no** keyword added, as shown:

```
-> vlan 5 no 802.1q 8
```

Note. The link aggregation group must be created first before it can be set to use 802.1Q tagging

For more specific information, see the [vlan 802.1q](#) command section in the *OmniSwitch 6250 CLI Reference Guide*.

Configuring the Frame Type

Once a port has been set to receive and send tagged frames, it will be able to receive or send tagged or untagged traffic. Tagged traffic will be subject to 802.1Q rules, while untagged traffic will behave as directed by normal switch operation. (Setting up rules for non-802.1Q traffic is defined in [Chapter 4, “Configuring VLANs.”](#)) A port can also be configured to accept only tagged frames.

To configure a port to only accept tagged frames, enter the **frame type** command at the CLI prompt:

```
-> vlan 802.1q 3/4 frame type tagged
```

To configure a port back to accepting both tagged and untagged traffic, use the same command with the **all** keyword, as shown:

```
-> vlan 802.1q 3/4 frame type all
```

Note. If you configure a port to accept only VLAN-tagged frames, then any frames received on this port that do not carry a VLAN identification (i.e., untagged frames or priority-tagged frames) will be discarded by the ingress rules for this port. Frames that are not discarded by this ingress rule are classified and processed according to the ingress rules for this port.

When a port is set to support both tagged and untagged traffic, multiple VLANs for 802.1Q traffic can be added to the port, but only one VLAN can be used to support untagged traffic. The untagged traffic VLAN will always be the port’s default VLAN.

Note. You cannot configure a link aggregation group to accept only tagged frames.

For more specific information, see the [vlan 802.1q frame type](#) command section in the *OmniSwitch 6250 CLI Reference Guide*.

Show 802.1Q Information

After configuring a port or link aggregation group to be a tagged port, you can view the settings by using the **show 802.1q** command, as demonstrated:

```
-> show 802.1q 3/4
```

```
Acceptable Frame Type : Any Frame Type
Force Tag Internal    : NA
```

```
Tagged VLANs      Internal Description
-----+-----+
          2      TAG PORT 3/4 VLAN 2
```

```
-> show 802.1q 2
```

```
Tagged VLANs      Internal Description
-----+-----+
          3      TAG AGGREGATE 2 VLAN 3
```

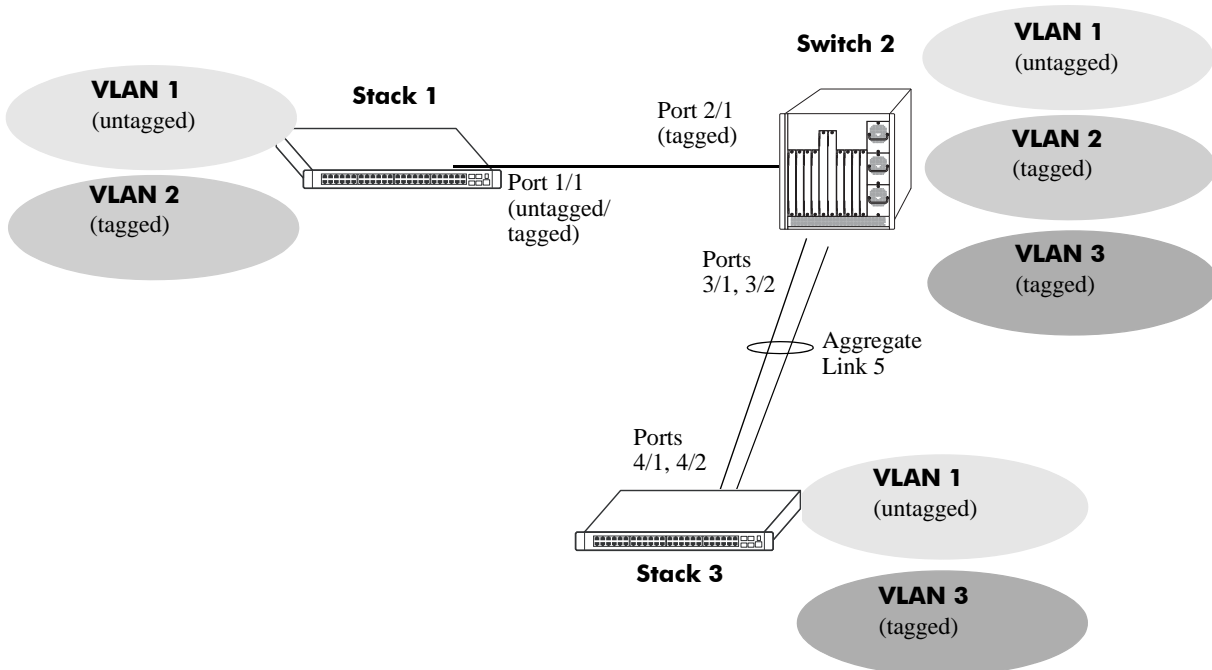
To display all VLANs, enter the following command:

```
-> show vlan port
```

Application Example

In this section the steps to create 802.1Q connections between switches are shown.

The following diagram shows a simple network employing 802.1Q on both regular ports and link aggregation groups.



802.1Q Application Example

The following sections show how to create the network illustrated above.

Connecting Stack 1 and Switch 2 Using 802.1Q

The following steps apply to Stack 1. They will attach port 1/1 to VLAN 2 and set the port to accept 802.1Q tagged traffic and untagged traffic.

- 1 Create VLAN 2 by entering `vlan 2` as shown below (VLAN 1 is the default VLAN for the switch):

```
-> vlan 2
```

- 2 Set port 1/1 as a tagged port and assign it to VLAN 2 by entering the following:

```
-> vlan 2 802.1q 1/1
```

- 3 Check the configuration by using the `show 802.1q` command as follows:

```
-> show 802.1q 1/1
```

```
Acceptable Frame Type : Any Frame Type
Force Tag Internal    : NA
```

```
Tagged VLANs      Internal Description
-----+-----+
          2      TAG PORT 1/1 VLAN 2
```

The following steps apply to Switch 2. They will attach port 2/1 to VLAN 2 and set the port to accept 802.1Q tagged traffic only:

- 1 Create VLAN 2 by entering **vlan 2** as shown below (VLAN 1 is the default VLAN for the switch):

```
-> vlan 2
```

- 2 Set port 2/1 as a tagged port and assign it to VLAN 2 by entering the following:

```
-> vlan 2 802.1q 2/1
```

- 3 Set port 2/1 to accept only tagged traffic by entering the following:

```
-> vlan 802.1q 2/1 frame type tagged
```

- 4 Check the configuration by using the **show 802.1q** command, as follows:

```
-> show 802.1q 2/1
```

```
Acceptable Frame Type   :          tagged only
Force Tag Internal      :                   NA
```

```
Tagged VLANs           Internal Description
-----+-----+-----+
                2          TAG PORT 2/1 VLAN 2
```

Connecting Switch 2 and Stack 3 Using 802.1Q

The following steps apply to Switch 2. They will attach ports 3/1 and 3/2 as link aggregation group 5 to VLAN 3.

- 1 Configure static aggregate VLAN 5 by entering the following:

```
-> static linkagg 5 size 2
```

- 2 Assign ports 3/1 and 3/2 to static aggregate VLAN 5 by entering the following two commands:

```
-> static agg 3/1 agg num 5
```

```
-> static agg 3/2 agg num 5
```

- 3 Create VLAN 3 by entering the following:

```
-> vlan 3
```

- 4 Configure 802.1Q tagging with a tagging ID of 3 on link aggregation group 5 (on VLAN 3) by entering **vlan 3 802.1q 5** as shown below:

```
-> vlan 3 802.1q 5
```

- 5 Check the configuration by using the **show 802.1q** command as follows:

```
-> show 802.1q 5
```

```
Tagged VLANs           Internal Description
-----+-----+-----+
                3          TAG AGGREGATE 5 VLAN 3
```

The following steps apply to Stack 3. They will attach ports 4/1 and 4/2 as link aggregation group 5 to VLAN 3.

- 1 Configure static link aggregation group 5 by entering the following:

```
-> static linkagg 5 size 2
```

- 2 Assign ports 4/1 and 4/2 to static link aggregation group 5 by entering the following two commands:

```
-> static agg 4/1 agg num 5  
-> static agg 4/2 agg num 5
```

- 3 Create VLAN 3 by entering the following:

```
-> vlan 3
```

- 4 Configure 802.1Q tagging with a tagging ID of 3 on static link aggregation group 5 (on VLAN 3) by entering the following:

```
-> vlan 3 802.1q 5
```

- 5 Check the configuration by using the **show 802.1q** command, as follows:

```
-> show 802.1q 5
```

```
Tagged VLANs      Internal Description  
-----+-----  
          3      TAG AGGREGATE 5 VLAN 3
```

Verifying 802.1Q Configuration

To display information about the ports configured to handle tagging, use the following show command:

show 802.1q Displays 802.1Q tagging information for a single port or a link aggregation group.

For more information about the resulting display, see [Chapter 6, “802.1Q Commands,”](#) in the *OmniSwitch 6250 CLI Reference Guide*.

23 Configuring Static Link Aggregation

Alcatel-Lucent's static link aggregation software allows you to combine several physical links into one large virtual link known as a link aggregation *group*. Using link aggregation provides the following benefits:

- **Scalability.** It is possible to configure up to 32 link aggregation groups that consist of 2, 4, or 8 10-Mbps, 100-Mbps, 1-Gbps, or 10-Gbps Ethernet links.
- **Reliability.** If one of the physical links in a link aggregate group goes down (unless it is the last one) the link aggregate group can still operate.
- **Ease of Migration.** Link aggregation can ease the transition from 100-Mbps Ethernet backbones to Gigabit Ethernet backbones.

In This Chapter

This chapter describes the basic components of static link aggregation and how to configure them through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Configuring static link aggregation groups on [page 23-7](#).
- Adding and deleting ports from a static aggregate group on [page 23-9](#).
- Modifying static link aggregation default values on [page 23-10](#).

Note. You can also configure and monitor static link aggregation with WebView, Alcatel-Lucent's embedded web-based device management application. WebView is an interactive and easy-to-use GUI that can be launched from OmniVista or a web browser. Please refer to WebView's online documentation for more information on configuring and monitoring static link aggregation with WebView.

Static Link Aggregation Specifications

The table below lists specifications for static groups.

Platforms Supported	OmniSwitch 6250 Series
Maximum number of link aggregation groups	32 (per switch or a stack of switches)
Number of links per group supported	2, 4, or 8 (per switch or a stack of switches)
Range for optional group name	1 to 255 characters
CLI Command Prefix Recognition	All static link aggregation configuration commands support prefix recognition. (Static link aggregation show commands do not support prefix recognition.) See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

Static Link Aggregation Default Values

The table below lists default values and the commands to modify them for static aggregate groups.

Parameter Description	Command	Default Value/Comments
Administrative State	<code>static linkagg admin state</code>	enabled
Group Name	<code>static linkagg name</code>	No name configured

Quick Steps for Configuring Static Link Aggregation

Follow the steps below for a quick tutorial on configuring a static aggregate link between two switches. Additional information on how to configure each command is given in the subsections that follow.

- 1 Create the static aggregate link on the local switch with the **static linkagg size** command. For example:

```
-> static linkagg 1 size 4
```

- 2 Assign all the necessary ports with the **static agg agg num** command. For example:

```
-> static agg 1/1 agg num 1  
-> static agg 1/2 agg num 1  
-> static agg 1/3 agg num 1  
-> static agg 1/4 agg num 1
```

- 3 Create a VLAN for this static link aggregate group with the **vlan** command. For example:

```
-> vlan 10 port default 1
```

- 4 Create the equivalent static aggregate link on the remote switch with the **static linkagg size** command. For example:

```
-> static linkagg 1 size 4
```

- 5 Assign all the necessary ports with the **static agg agg num** command. For example:

```
-> static agg 1/9 agg num 1  
-> static agg 1/10 agg num 1  
-> static agg 1/11 agg num 1  
-> static agg 1/12 agg num 1
```

- 6 Create a VLAN for this static link aggregate group with the **vlan** command. For example:

```
-> vlan 10 port default 1
```

Note. *Optional.* You can verify your static link aggregation settings with the **show linkagg** command. For example:

```
-> show linkagg 1
Static Aggregate
SNMP Id           : 40000001,
Aggregate Number  : 1,
SNMP Descriptor   : Omnichannel Aggregate Number 1 ref 40000001 size 4,
Name              : ,
Admin State       : ENABLED,
Operational State : UP,
Aggregate Size    : 4,
Number of Selected Ports : 4,
Number of Reserved Ports : 4,
Number of Attached Ports : 4,
Primary Port      : 1/1
```

You can also use the **show linkagg port** port command to display information on specific ports. See [“Displaying Static Link Aggregation Configuration and Statistics”](#) on page 23-12 for more information on the **show** commands.

An example of what these commands look like entered sequentially on the command line on the local switch:

```
-> static linkagg 1 size 4
-> static agg 1/1 agg num 1
-> static agg 1/2 agg num 1
-> static agg 1/3 agg num 1
-> static agg 1/4 agg num 1
-> vlan 10 port default 1
```

And an example of what these commands look like entered sequentially on the command line on the remote switch:

```
-> static linkagg 1 size 4
-> static agg 1/9 agg num 1
-> static agg 1/10 agg num 1
-> static agg 1/11 agg num 1
-> static agg 1/12 agg num 1
-> vlan 10 port default 1
```

Static Link Aggregation Overview

Link aggregation allows you to combine 2, 4, or 8 physical connections into large virtual connections known as link aggregation *groups*. You can configure up to 32 link aggregation groups per a standalone switch or a stack of switches. Each group can consist of 2, 4, or 8 10-Mbps, 100-Mbps, 1-Gbps, or 10-Gbps Ethernet links.

You can create Virtual LANs (VLANs), 802.1Q framing, configure Quality of Service (QoS) conditions, and other networking features on link aggregation groups because the switch's software treats these virtual links just like physical links. (See [“Relationship to Other Features” on page 23-6](#) for more information on how link aggregation interacts with other software features.)

Load balancing for Layer 2 non-IP packets is on a MAC address basis and for IP packets the balancing algorithm uses IP address as well. Ports *must* be of the same speed within the same link aggregate group.

Alcatel-Lucent's link aggregation software allows you to configure the following two different types of link aggregation groups:

- Static link aggregate groups
- Dynamic link aggregate groups

This chapter describes static link aggregation. For information on dynamic link aggregation, please refer to [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Static Link Aggregation Operation

Static link aggregate groups are virtual links between two nodes consisting of 2, 4, or 8 10-Mbps, 100-Mbps, or 1-or 10-Gbps fixed physical links. You can configure up to 32 link aggregation groups per a standalone switch or a stack of switches.

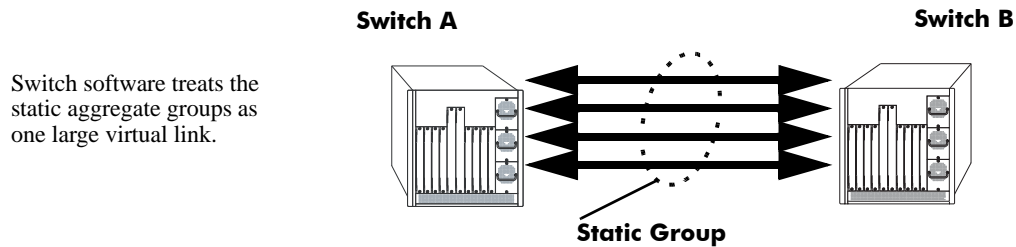
Static aggregate groups can be created between each of the following OmniSwitch products:

- two OmniSwitch 6250 switches.
- an OmniSwitch 6250 switch and an OmniSwitch 7700/7800, OmniSwitch 8800, or OmniSwitch 6600 Series switch.
- an OmniSwitch 6250 switch and an early-generation Alcatel-Lucent switch, such as an Omni Switch/Router.

Note. Static aggregate groups cannot be created between an OmniSwitch 6250 switch and some switches from other vendors.

The figure below shows a static aggregate group that has been configured between Switch A and Switch B. The static aggregate group links four ports on a single OS9-GNI-C24 on Switch A to two ports on one

OS9-GNI-C24 and two ports on another OS9-GNI-C24 on Switch B. The network administrator has created a separate VLAN for this group so users can use this high speed link.



Example of a Static Link Aggregate Group Network

See [“Configuring Static Link Aggregation Groups” on page 23-7](#) for information on using Command Line Interface (CLI) commands to configure static aggregate groups and see [“Displaying Static Link Aggregation Configuration and Statistics” on page 23-12](#) for information on using CLI to monitor static aggregate groups.

Relationship to Other Features

Link aggregation groups are supported by other switch software features. The following features have CLI commands or command parameters that support link aggregation:

- **VLANs.** For more information on VLANs see [Chapter 4, “Configuring VLANs.”](#)
- **802.1Q.** For more information on configuring and monitoring 802.1Q see [Chapter 22, “Configuring 802.1Q.”](#)
- **Spanning Tree.** For more information on Spanning Tree see [Chapter 23, “Configuring Static Link Aggregation.”](#)

Note. See [“Application Example” on page 23-11](#) for tutorials on using link aggregation with other features.

Configuring Static Link Aggregation Groups

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to configure static link aggregate groups. See [“Configuring Mandatory Static Link Aggregate Parameters” on page 23-7](#) for more information.

Note. See [“Quick Steps for Configuring Static Link Aggregation” on page 23-3](#) for a brief tutorial on configuring these mandatory parameters.

Alcatel-Lucent's link aggregation software is preconfigured with the default values for static aggregate groups as shown in the table in [“Static Link Aggregation Default Values” on page 23-2](#). If you need to modify any of these parameters, please see [“Modifying Static Aggregation Group Parameters” on page 23-10](#) for more information.

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of CLI commands for link aggregation.

Configuring Mandatory Static Link Aggregate Parameters

When configuring static link aggregates on a switch you must perform the following steps:

- 1 Create the Static Aggregate Group on the Local and Remote Switches.** To create a static aggregate group use the **static linkagg size** command, which is described in [“Creating and Deleting a Static Link Aggregate Group” on page 23-8](#).
- 2 Assign Ports on the Local and Remote Switches to the Static Aggregate Group.** To assign ports to the static aggregate group you use the **static agg agg num** command, which is described in [“Adding and Deleting Ports in a Static Aggregate Group” on page 23-9](#).

Note. Depending on the needs of your network you may need to configure additional parameters. Commands to configure optional static aggregate parameters are described in [“Modifying Static Aggregation Group Parameters” on page 23-10](#).

Creating and Deleting a Static Link Aggregate Group

The following subsections describe how to create and delete static link aggregate groups with the **static linkagg size** command.

Creating a Static Aggregate Group

You can create up to 32 static and/or dynamic link aggregation groups per a standalone switch or a stack of switches. To create a static aggregate group on a switch, enter **static linkagg** followed by the user-specified aggregate number (which can be 0 through 31), **size**, and the number of links in the static aggregate group, which can be 2, 4, or 8.

For example, to create static aggregate group 5 that consists of eight links, on a switch, you would enter:

```
-> static linkagg 5 size 8
```

Note. The number of links assigned to a static aggregate group should always be close to the number of physical links that you plan to use. For example, if you are planning to use 2 physical links you should create a group with a size of 2 and not 4 or 8.

As an option you can also specify a name and/or the administrative status of the group by entering **static linkagg** followed by the user-specified aggregate number, **size**, the number of links in the static aggregate group, **name**, the optional name (which can be up to 255 characters long), **admin state**, and either **enable** or **disable** (the default is **enable**).

For example, to create static aggregate group 5 called “static1” consisting of eight links that is administratively disabled enter:

```
-> static linkagg 5 size 8 name static1 admin state disable
```

Note. If you want to specify spaces within a name for a static aggregate group the name must be specified within quotes (e.g., “Static Aggregate Group 5”).

Deleting a Static Aggregate Group

To delete a static aggregation group from a switch use the **no** form of the **static linkagg size** command by entering **no static linkagg** followed by the number that identifies the group. For example, to remove static aggregate group 5 from a switch’s configuration you would enter:

```
-> no static linkagg 5
```

Note. You must delete any attached ports with the **static agg agg num** command before you can delete a static link aggregate group.

Adding and Deleting Ports in a Static Aggregate Group

The following subsections describe how to add and delete ports in a static aggregate group with the **static agg agg num** command.

Adding Ports to a Static Aggregate Group

The number of ports assigned in a static aggregate group can be less than or equal to the maximum size you specified in the **static linkagg size** command. To assign a port to a static aggregate group you use the **static agg agg num** command by entering **static agg** followed by the slot number, a slash (/), the port number, **agg num**, and the number of the static aggregate group. Ports must be of the same speed (i.e., all 10 Mbps, all 100 Mbps, or all 1 Gbps).

For example, to assign ports 1, 2, and 3 in slot 1 to static aggregate group 10 (which has a size of 4) you would enter:

```
-> static agg 1/1 agg num 10
-> static agg 1/2 agg num 10
-> static agg 1/3 agg num 10
```

Note. A port may belong to only one aggregate group. In addition, mobile ports cannot be aggregated. See [Chapter 6, “Assigning Ports to VLANs,”](#) for more information on mobile ports.

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to assign port 1 in slot 1 to static aggregate group 10 and document that port 1 in slot 5 is a Giga Ethernet port you would enter:

```
-> static gigaethernet agg 1/1 agg num 10
```

Note. The **ethernet**, **fastethernet**, and **gigaethernet** keywords do not modify a port's configuration. See [Chapter 23, “Configuring Static Link Aggregation,”](#) for information on configuring Ethernet ports.

Removing Ports from a Static Aggregate Group

To remove a port from a static aggregate group you use the **no** form of the **static agg agg num** command by entering **static agg no** followed by the slot number, a slash (/), and the port number. For example, to remove port 4 in slot 1 from a static aggregate group you would enter:

```
-> static agg no 1/4
```

Ports must be deleted in the reverse order in which they were assigned. For example, if port 9 through 16 were assigned to static aggregate group 2 you must first delete port 16, then port 15, and so forth. The following is an example of how to delete ports in the proper sequence from the console:

```
-> static agg no 1/24
-> static agg no 1/23
-> static agg no 1/22
```

Modifying Static Aggregation Group Parameters

This section describes how to modify the following static aggregate group parameters:

- Static aggregate group name (see “[Modifying the Static Aggregate Group Name](#)” on page 23-10)
- Static aggregate group administrative state (see “[Modifying the Static Aggregate Group Administrative State](#)” on page 23-10)

Modifying the Static Aggregate Group Name

The following subsections describe how to modify the name of the static aggregate group with the **static linkagg name** command.

Creating a Static Aggregate Group Name

To create a name for a static aggregate group by entering **static linkagg** followed by the number of the static aggregate group, **name**, and the user-specified name of the group, which can be up to 255 characters long. For example, to configure static aggregate group 4 with the name “Finance” you would enter:

```
-> static linkagg 4 name Finance
```

Note. If you want to specify spaces within a name for a static aggregate group the name must be specified within quotes (e.g., “Static Aggregate Group 4”).

Deleting a Static Aggregate Group Name

To remove a name from a static aggregate group you use the **no** form of the **static linkagg name** command by entering **static linkagg** followed by the number of the static aggregate group and **no name**. For example, to remove any user-specified name from static aggregate group 4 you would enter:

```
-> static linkagg 4 no name
```

Modifying the Static Aggregate Group Administrative State

By default, the administrative state for a static aggregate group is enabled. The following subsections describe how to enable and disable the administrative state with the **static linkagg admin state** command.

Enabling the Static Aggregate Group Administrative State

To enable a static aggregate group by entering **static linkagg** followed by the number of the group and **admin state enable**. For example, to enable static aggregate group 1 you would enter:

```
-> static linkagg 1 admin state enable
```

Disabling the Static Aggregate Group Administrative State

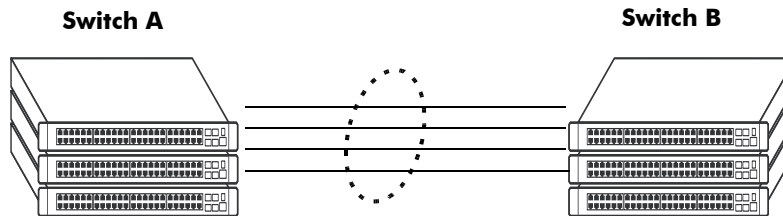
To disable a static aggregate group by entering **static linkagg** followed by the number of the group and **admin state disable**. For example, to disable static aggregate group 1 you would enter:

```
-> static linkagg 1 admin state disable
```


Application Example

Static link aggregation groups are treated by the switch's software the same way it treats individual physical ports. This section demonstrates this by providing a sample network configuration that uses static link aggregation along with other software features. In addition, a tutorial is provided that shows how to configure this sample network using Command Line Interface (CLI) commands.

The figure below shows VLAN 8, which has been configured on static aggregate 1 and uses 802.1Q tagging. The actual physical links connect ports 4/1, 4/2, 4/3, and 4/4 on Switch A to port 2/41, 2/42, 2/43, and 2/44 on Switch B.



Static Aggregate Group 1
VLAN 8 with 802.1Q tagging has been configured to use this group.

Sample Network Using Static Link Aggregation

Follow the steps below to configure this network:

Note. Only the steps to configure the local (i.e., Switch A) switch are provided here since the steps to configure the remote (i.e., Switch B) switch would not be significantly different.

- 1 Configure static aggregate group 1 by entering **static linkagg 1 size 4** as shown below:

```
-> static linkagg 1 size 4
```

- 2 Assign ports 4/1, 4/2, 4/3, and 4/4 to static aggregate group 1 by entering:

```
-> static agg 4/1 agg num 1
-> static agg 4/2 agg num 1
-> static agg 4/3 agg num 1
-> static agg 4/4 agg num 1
```

- 3 Create VLAN 8 by entering:

```
-> vlan 8
```

- 4 Configure 802.1Q tagging with a tagging ID of 8 on static aggregate group 1 (on VLAN 8) by entering:

```
-> vlan 8 802.1q 1
```

- 5 Repeat steps 1 through 4 on Switch B. All the commands would be the same except you would substitute the appropriate port numbers.

Note. *Optional.* Use the [show 802.1q](#) command to display 802.1Q configurations.

Displaying Static Link Aggregation Configuration and Statistics

You can use Command Line Interface (CLI) **show** commands to display the current configuration and statistics of link aggregation. These commands include the following:

- show linkagg** Displays information on link aggregation groups.
- show linkagg port** Displays information on link aggregation ports.

When you use the **show linkagg** command without specifying the link aggregation group number and when you use the **show linkagg port** command without specifying the slot and port number these commands provide a “global” view of switch-wide link aggregate group and link aggregate port information, respectively.

For example, to display global statistics on all link aggregate groups (both static and dynamic) you would enter:

```
-> show linkagg
```

A screen similar to the following would be displayed:

Number	Aggregate	SNMP Id	Size	Admin State	Oper State	Att/Sel Ports
1	Static	40000001	8	ENABLED	UP	2 2
2	Dynamic	40000002	4	ENABLED	DOWN	0 0
3	Dynamic	40000003	8	ENABLED	DOWN	0 2
4	Static	40000005	2	DISABLED	DOWN	0 0

When you use the **show linkagg** command with the link aggregation group number and when you use the **show linkagg port** command with the slot and port number these commands provide detailed views of link aggregate group and link aggregate port information, respectively. These detailed views provide excellent tools for diagnosing and troubleshooting problems.

For example, to display detailed statistics for port 1 in slot 4 that is attached to static link aggregate group 1 you would enter:

```
-> show linkagg port 4/1
```

A screen similar to the following would be displayed:

```
Static Aggregable Port
SNMP Id                : 4001,
Slot/Port              : 4/1,
Administrative State   : ENABLED,
Operational State     : DOWN,
Port State             : CONFIGURED,
Link State             : DOWN,
Selected Agg Number   : 2,
Port position in the aggregate : 0,
Primary port          : NONE
```

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of **show** commands for link aggregation.

24 Configuring Dynamic Link Aggregation

Alcatel-Lucent's dynamic link aggregation software allows you to combine several physical links into one large virtual link known as a link aggregation *group*. Using link aggregation provides the following benefits:

- **Scalability.** It is possible to configure up to 32 link aggregation groups that consist of 2, 4, or 8 10-Mbps, 100-Mbps, 1-Gbps, or 10-Gbps Ethernet links.
- **Reliability.** If one of the physical links in a link aggregate group goes down (unless it is the last one) the link aggregate group can still operate.
- **Ease of Migration.** Link aggregation can ease the transition from 100-Mbps Ethernet backbones to Gigabit Ethernet backbones.

In This Chapter

This chapter describes the basic components of dynamic link aggregation and how to configure them through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Configuring dynamic link aggregation groups on [page 24-10](#).
- Configuring ports so they can be aggregated in dynamic link aggregation groups on [page 24-12](#).
- Modifying dynamic link aggregation parameters on [page 24-14](#).

Note. You can also configure and monitor dynamic link aggregation with WebView, Alcatel-Lucent's embedded Web-based device management application. WebView is an interactive and easy-to-use GUI that can be launched from OmniVista or a Web browser. Please refer to WebView's online documentation for more information on configuring and monitoring dynamic link aggregation with WebView.

Dynamic Link Aggregation Specifications

The table below lists specifications for dynamic aggregation groups and ports:

IEEE Specifications Supported	802.3ad — Aggregation of Multiple Link Segments
Platforms Supported	OmniSwitch 6250 Series
Maximum number of link aggregation groups	32 (per standalone switch or a stack of switches)
Range for optional group name	1 to 255 characters
Number of links per group supported	2, 4, or 8
Group actor admin key	0 to 65535
Group actor system priority	0 to 65535
Group partner system priority	0 to 65535
Group partner admin key	0 to 65535
Port actor admin key	0 to 65535
Port actor system priority	0 to 255
Port partner admin key	0 to 65535
Port partner admin system priority	0 to 255
Port actor port	0 to 65535
Port actor port priority	0 to 255
Port partner admin port	0 to 65535
Port partner admin port priority	0 to 255
CLI Command Prefix Recognition	All dynamic link aggregation configuration commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

Dynamic Link Aggregation Default Values

The table below lists default values for dynamic aggregate groups.

Parameter Description	Command	Default Value/Comments
Group Administrative State	lACP linkagg admin state	enabled
Group Name	lACP linkagg name	No name configured
Group Actor Administrative Key	lACP linkagg actor admin key	0
Group Actor System Priority	lACP linkagg actor system priority	0
Group Actor System ID	lACP linkagg actor system id	00:00:00:00:00:00
Group Partner System ID	lACP linkagg partner system id	00:00:00:00:00:00
Group Partner System Priority	lACP linkagg partner system priority	0
Group Partner Administrative Key	lACP linkagg partner admin key	0
Actor Port Administrative State	lACP agg actor admin state	active timeout aggregate
Actor Port System ID	lACP agg actor system id	00:00:00:00:00:00
Partner Port System Administrative State	lACP agg partner admin state	active timeout aggregate
Partner Port Admin System ID	lACP agg partner admin system id	00:00:00:00:00:00
Partner Port Administrative Key	lACP agg partner admin key	0
Partner Port Admin System Priority	lACP agg partner admin system priority	0
Actor Port Priority	lACP agg actor port priority	0
Partner Port Administrative Port	lACP agg partner admin port	0
Partner Port Priority	lACP agg partner admin port priority	0

Quick Steps for Configuring Dynamic Link Aggregation

Follow the steps below for a quick tutorial on configuring a dynamic aggregate link between two switches. Additional information on how to configure each command is given in the subsections that follow.

1 Create the dynamic aggregate group on the local (actor) switch with the **lacp linkagg size** command as shown below:

```
-> lacp linkagg 2 size 8 actor admin key 5
```

2 Configure ports (the number of ports should be less than or equal to the size value set in step 1) with the same actor administrative key (which allows them to be aggregated) with the **lacp agg actor admin key** command. For example:

```
-> lacp agg 1/1 actor admin key 5
-> lacp agg 1/4 actor admin key 5
-> lacp agg 3/3 actor admin key 5
-> lacp agg 5/4 actor admin key 5
-> lacp agg 6/1 actor admin key 5
-> lacp agg 6/2 actor admin key 5
-> lacp agg 7/3 actor admin key 5
-> lacp agg 8/1 actor admin key 5
```

3 Create a VLAN for this dynamic link aggregate group with the **vlan** command. For example:

```
-> vlan 2 port default 2
```

4 Create the equivalent dynamic aggregate group on the remote (partner) switch with the **lacp linkagg size** command as shown below:

```
-> lacp linkagg 2 size 8 actor admin key 5
```

5 Configure ports (the number of ports should be less than or equal to the size value set in step 4) with the same actor administrative key (which allows them to be aggregated) with the **lacp agg actor admin key** command. For example:

```
-> lacp agg 2/1 actor admin key 5
-> lacp agg 3/1 actor admin key 5
-> lacp agg 3/3 actor admin key 5
-> lacp agg 3/6 actor admin key 5
-> lacp agg 5/1 actor admin key 5
-> lacp agg 5/6 actor admin key 5
-> lacp agg 8/1 actor admin key 5
-> lacp agg 8/3 actor admin key 5
```

6 Create a VLAN for this dynamic link aggregate group with the **vlan** command. For example:

```
-> vlan 2 port default 2
```

Note. As an option, you can verify your dynamic aggregation group settings with the **show linkagg** command on either the actor or the partner switch. For example:

```
-> show linkagg 2
Dynamic Aggregate
  SNMP Id           : 40000002,
  Aggregate Number  : 2,
  SNMP Descriptor   : Dynamic Aggregate Number 2 ref 40000002 size 8,
  Name              : ,
  Admin State       : ENABLED,
  Operational State : UP,
  Aggregate Size    : 8,
  Number of Selected Ports : 8,
  Number of Reserved Ports : 8,
  Number of Attached Ports : 8,
  Primary Port      : 1/1,
LACP
  MACAddress        : [00:1f:cc:00:00:00],
  Actor System Id   : [00:20:da:81:d5:b0],
  Actor System Priority : 0,
  Actor Admin Key   : 5,
  Actor Oper Key    : 0,
  Partner System Id : [00:20:da:81:d5:b1],
  Partner System Priority : 0,
  Partner Admin Key : 5,
  Partner Oper Key  : 0
```

You can also use the **show linkagg port** port command to display information on specific ports. See [“Displaying Dynamic Link Aggregation Configuration and Statistics” on page 24-32](#) for more information on **show** commands.

An example of what these commands look like entered sequentially on the command line on the actor switch:

```
-> lacp linkagg 2 size 8 actor admin key 5
-> lacp agg 1/1 actor admin key 5
-> lacp agg 1/4 actor admin key 5
-> lacp agg 3/3 actor admin key 5
-> lacp agg 5/4 actor admin key 5
-> lacp agg 6/1 actor admin key 5
-> lacp agg 6/2 actor admin key 5
-> lacp agg 7/3 actor admin key 5
-> lacp agg 8/1 actor admin key 5
-> vlan 2 port default 2
```

An example of what these commands look like entered sequentially on the command line on the partner switch:

```
-> lacp linkagg 2 size 8 actor admin key 5
-> lacp agg 2/1 actor admin key 5
-> lacp agg 3/1 actor admin key 5
-> lacp agg 3/3 actor admin key 5
-> lacp agg 3/6 actor admin key 5
-> lacp agg 5/1 actor admin key 5
-> lacp agg 5/6 actor admin key 5
-> lacp agg 8/1 actor admin key 5
-> lacp agg 8/3 actor admin key 5
-> vlan 2 port default 2
```


Dynamic Link Aggregation Overview

Link aggregation allows you to combine 2, 4, or 8 physical connections into large virtual connections known as link aggregation *groups*. You can configure up to 32 link aggregation groups per a standalone switch or a stack of switches. Each group can consist of 2, 4, or 8 10-Mbps, 100-Mbps, 1-Gbps, or 10-Gbps Ethernet links.

You can create Virtual LANs (VLANs), 802.1Q framing, configure Quality of Service (QoS) conditions, and other networking features on link aggregation groups because switch software treats these virtual links just like physical links. (See “[Relationship to Other Features](#)” on page 24-9 for more information on how link aggregation interacts with other software features.)

Link aggregation groups are identified by unique MAC addresses, which are created by the switch but can be modified by the user at any time. Load balancing for Layer 2 non-IP packets is on a MAC address basis and for IP packets the balancing algorithm uses the IP address as well. Ports *must* be of the same speed within the same aggregate group.

Alcatel-Lucent’s link aggregation software allows you to configure the following two different types of link aggregation groups:

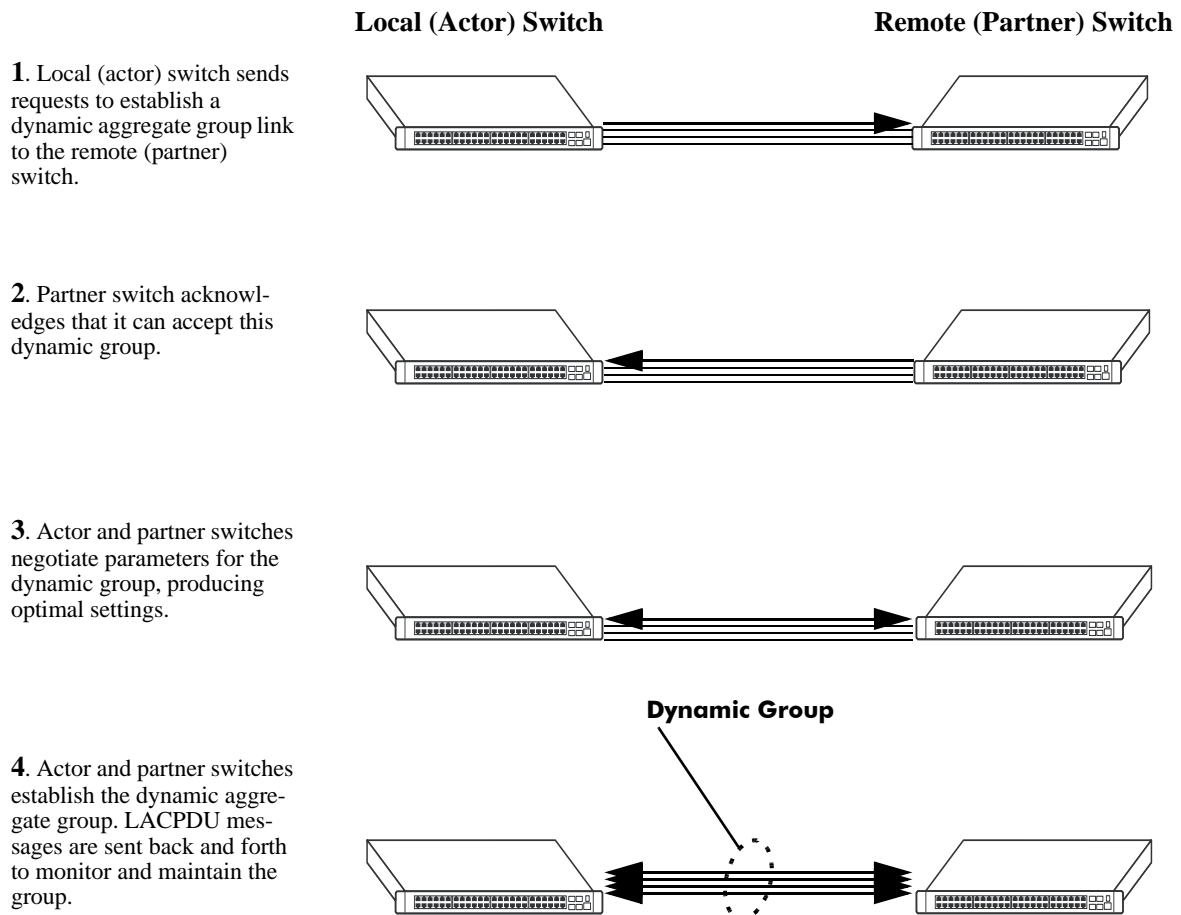
- Static link aggregate groups
- Dynamic link aggregate groups

This chapter describes dynamic link aggregation. For information on static link aggregation, please refer to [Chapter 23, “Configuring Static Link Aggregation.”](#)

Dynamic Link Aggregation Operation

Dynamic aggregate groups are virtual links between two nodes consisting of 2, 4, or 8 10-Mbps, 100-Mbps, or 1-or 10-Gbps fixed physical links. Dynamic aggregate groups use the standard IEEE 802.3ad Link Aggregation Control Protocol (LACP) to dynamically establish the best possible configuration for the group. This task is accomplished by special Link Aggregation Control Protocol Data Unit (LACPDU) frames that are sent and received by switches on both sides of the link to monitor and maintain the dynamic aggregate group.

The figure on the following page shows a dynamic aggregate group that has been configured between Switch A and Switch B. The dynamic aggregate group links four ports on Switch A to four ports on Switch B.



Example of a Dynamic Aggregate Group Network

Dynamic aggregate groups can be created between each of the following OmniSwitch products:

- two OmniSwitch 6250 switches.
- an OmniSwitch 6250 switch and an OmniSwitch 7700/7800, OmniSwitch 8800, or OmniSwitch 6600 Series switch.
- an OmniSwitch 6250 switch and an early-generation Alcatel-Lucent switch, such as an Omni Switch/Router.
- an OmniSwitch 6250 switch and another vendor's switch if that vendor supports IEEE 802.3ad LACP.

See [“Configuring Dynamic Link Aggregate Groups”](#) on page 24-10 for information on using Command Line Interface (CLI) commands to configure dynamic aggregate groups and see [“Displaying Dynamic Link Aggregation Configuration and Statistics”](#) on page 24-32 for information on using the CLI to monitor dynamic aggregate groups.

Relationship to Other Features

Link aggregation groups are supported by other switch software features. For example, you can configure 802.1Q tagging on link aggregation groups in addition to configuring it on individual ports. The following features have CLI commands or command parameters that support link aggregation:

- **VLANs.** For more information on VLANs, see [Chapter 4, “Configuring VLANs.”](#)
- **802.1Q.** For more information on configuring and monitoring 802.1Q, see [Chapter 22, “Configuring 802.1Q.”](#)
- **Spanning Tree.** For more information on Spanning Tree, see [Chapter 11, “Configuring Spanning Tree Parameters.”](#)

Note. See [“Application Examples” on page 24-29](#) for tutorials on using link aggregation with other features.

Configuring Dynamic Link Aggregate Groups

This section describes how to use Alcatel-Lucent's Command Line Interface (CLI) commands to create, modify, and delete dynamic aggregate groups. See [“Configuring Mandatory Dynamic Link Aggregate Parameters” on page 24-10](#) for more information.

Note. See [“Quick Steps for Configuring Dynamic Link Aggregation” on page 24-4](#) for a brief tutorial on configuring these mandatory parameters.

Alcatel-Lucent's link aggregation software is preconfigured with the default values for dynamic aggregate groups and ports shown in the table in [“Dynamic Link Aggregation Default Values” on page 24-3](#). For most configurations, using only the steps described in [“Creating and Deleting a Dynamic Aggregate Group” on page 24-11](#) will be necessary to configure a dynamic link aggregate group. However, if you need to modify any of the parameters listed in the table on [page 24-3](#), please see [“Modifying Dynamic Link Aggregate Group Parameters” on page 24-14](#) for more information.

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of **show** commands for link aggregation.

Configuring Mandatory Dynamic Link Aggregate Parameters

When configuring LACP link aggregates on a switch you must perform the following steps:

- 1 Create the Dynamic Aggregate Groups on the Local (Actor) and Remote (Partner) Switches.** To create a dynamic aggregate group use the **lacp linkagg size** command, which is described in [“Creating and Deleting a Dynamic Aggregate Group” on page 24-11](#).
- 2 Configure the Same Administrative Key on the Ports You Want to Join the Dynamic Aggregate Group.** To configure ports with the same administrative key (which allows them to be aggregated), use the **lacp agg actor admin key** command, which is described in [“Configuring Ports to Join and Removing Ports in a Dynamic Aggregate Group” on page 24-12](#).

Note. Depending on the needs of your network you may need to configure additional parameters. Commands to configure optional dynamic link aggregate parameters are described in [“Modifying Dynamic Link Aggregate Group Parameters” on page 24-14](#). These commands must be executed after you create a dynamic aggregate group.

Creating and Deleting a Dynamic Aggregate Group

The following subsections describe how to create and delete dynamic aggregate groups with the **lacp linkagg size** command.

Creating a Dynamic Aggregate Group

To configure a dynamic aggregate group, enter **lacp linkagg** followed by the user-configured dynamic aggregate number (which can be from 0 to 31), **size**, and the maximum number of links that will belong to this dynamic aggregate group, which can be 2, 4, or 8. For example, to configure the dynamic aggregate group 2 consisting of eight links enter:

```
-> lacp linkagg 2 size 8
```

You can create up to 32 link aggregation (both static and dynamic) groups per a standalone switch or a stack of switches. In addition, you can also specify optional parameters shown in the table below. These parameters must be entered after **size** and the user-specified number of links.

lacp linkagg size keywords

name	admin state enable	partner admin key
actor system priority	admin state disable	actor admin key
partner system priority	actor system id	partner system id

For example, Alcatel-Lucent recommends assigning the actor admin key when you create the dynamic aggregate group to help ensure that ports are assigned to the correct group. To create a dynamic aggregate group with aggregate number 3 consisting of two ports with an admin actor key of 10, for example, enter:

```
-> lacp linkagg 3 size 2 actor admin key 10
```

Note. The optional keywords for this command may be entered in any order as long as they are entered after **size** and the user-specified number of links.

Deleting a Dynamic Aggregate Group

To remove a dynamic aggregation group configuration from a switch use the **no** form of the **lacp linkagg size** command by entering **no lacp linkagg** followed by its dynamic aggregate group number.

For example, to delete dynamic aggregate group 2 from a switch's configuration you would enter:

```
-> no lacp linkagg 2
```

Note. You cannot delete a dynamic aggregate group if it has any attached ports. To remove attached ports you must disable the dynamic aggregate group with the **lacp linkagg admin state** command, which is described in [“Disabling a Dynamic Aggregate Group”](#) on page 24-15.

Configuring Ports to Join and Removing Ports in a Dynamic Aggregate Group

The following subsections describe how to configure ports with the same administrative key (which allows them to be aggregated) or to remove them from a dynamic aggregate group with the **lACP agg actor admin key** command.

Configuring Ports To Join a Dynamic Aggregate Group

To configure ports with the same administrative key (which allows them to be aggregated) enter **lACP agg** followed by the slot number, a slash (/), the port number, **actor admin key**, and the user-specified actor administrative key (which can range from 0 to 65535). Ports must be of the same speed (i.e., all 10 Mbps, all 100 Mbps, or all 1 Gbps).

For example, to configure ports 1, 2, and 3 in slot 4 with an administrative key of 10 you would enter:

```
-> lACP agg 4/1 actor admin key 10
-> lACP agg 4/2 actor admin key 10
-> lACP agg 4/3 actor admin key 10
```

Note. A port may belong to only one aggregate group. In addition, mobile ports cannot be aggregated. See [Chapter 6, “Assigning Ports to VLANs,”](#) for more information on mobile ports.

You must execute the **lACP agg actor admin key** command on all ports in a dynamic aggregate group. If not, the ports will be unable to join the group.

In addition, you can also specify optional parameters shown in the table below. These keywords must be entered after the actor admin key and the user-specified actor administrative key value.

lACP agg actor admin key keywords

actor admin state	partner admin state	actor system id
actor system priority	partner admin system id	partner admin key
partner admin system priority	actor port priority	partner admin port
partner admin port priority		

Note. The **actor admin state** and **partner admin state** keywords have additional parameters, which are described in [“Modifying the Actor Port System Administrative State”](#) on page 24-19 and [“Modifying the Partner Port System Administrative State”](#) on page 24-23, respectively.

All of the optional keywords listed above for this command may be entered in any order as long as they appear after the **actor admin key** keywords and their user-specified value.

For example, to configure actor administrative key of 10, a local system ID (MAC address) of 00:20:da:06:ba:d3, and a local priority of 65535 to slot 4 port 1, enter:

```
-> lACP agg 4/1 actor admin key 10 actor system id 00:20:da:06:ba:d3 actor
system priority 65535
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to configure an actor administrative key of 10 and to document that the port is a 10-Mbps Ethernet port to slot 4 port 1, enter:

```
-> lacp agg ethernet 4/1 actor admin key 10
```

Note. The **ethernet**, **fastethernet**, and **gigaethernet** keywords do not modify a port's configuration. See [Chapter 1, "Configuring Ethernet Ports,"](#) for information on configuring Ethernet ports.

Removing Ports from a Dynamic Aggregate Group

To remove a port from a dynamic aggregate group, use the **no** form of the **lacp agg actor admin key** command by entering **lacp agg no** followed by the slot number, a slash (/), and the port number.

For example, to remove port 4 in slot 4 from any dynamic aggregate group you would enter:

```
-> lacp agg no 4/4
```

Ports must be deleted in the reverse order in which they were configured. For example, if port 9 through 16 were configured to join dynamic aggregate group 2 you must first delete port 16, then port 15, and so forth. The following is an example of how to delete ports in the proper sequence from the console:

```
-> lacp agg no 4/24  
-> lacp agg no 4/23  
-> lacp agg no 4/22
```

Modifying Dynamic Link Aggregate Group Parameters

The table on [page 24-3](#) lists default group and port settings for Alcatel-Lucent's dynamic link aggregation software. These parameters ensure compliance with the IEEE 802.3ad specification. For most networks, these default values do not need to be modified or will be modified automatically by switch software. However, if you need to modify any of these default settings see the following sections to modify parameters for:

- Dynamic aggregate groups beginning on [page 24-14](#)
- Dynamic aggregate actor ports beginning on [page 24-18](#)
- Dynamic aggregate partner ports beginning on [page 24-23](#)

Note. You *must* create a dynamic aggregate group before you can modify group or port parameters. See [“Configuring Dynamic Link Aggregate Groups” on page 24-10](#) for more information.

Modifying Dynamic Aggregate Group Parameters

This section describes how to modify the following dynamic aggregate group parameters:

- Group name (see [“Modifying the Dynamic Aggregate Group Name” on page 24-14](#))
- Group administrative state (see [“Modifying the Dynamic Aggregate Group Administrative State” on page 24-15](#))
- Group local (actor) switch actor administrative key (see [“Configuring and Deleting the Dynamic Aggregate Group Actor Administrative Key” on page 24-15](#))
- Group local (actor) switch system priority (see [“Modifying the Dynamic Aggregate Group Actor System Priority” on page 24-16](#))
- Group local (actor) switch system ID (see [“Modifying the Dynamic Aggregate Group Actor System ID” on page 24-16](#))
- Group remote (partner) administrative key (see [“Modifying the Dynamic Aggregate Group Partner Administrative Key” on page 24-17](#))
- Group remote (partner) system priority (see [“Modifying the Dynamic Aggregate Group Partner System Priority” on page 24-17](#))
- Group remote (partner) switch system ID (see [“Modifying the Dynamic Aggregate Group Partner System ID” on page 24-18](#))

Modifying the Dynamic Aggregate Group Name

The following subsections describe how to configure and remove a dynamic aggregate group name with the [**lacp linkagg name**](#) command.

Configuring a Dynamic Aggregate Group name

To configure a dynamic aggregate group name, enter **lacp linkagg** followed by the dynamic aggregate group number, **name**, and the user-specified name, which can be from 1 to 255 characters long.

For example, to name dynamic aggregate group 4 “Engineering” you would enter:

```
-> lacp linkagg 4 name Engineering
```

Note. If you want to specify spaces within a name, the name must be enclosed in quotes. For example:

```
-> lacp linkagg 4 name "Engineering Lab"
```

Deleting a Dynamic Aggregate Group Name

To remove a dynamic aggregate group name from a switch’s configuration use the **no** form of the **lacp linkagg name** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no name**.

For example, to remove any user-configured name from dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 no name
```

Modifying the Dynamic Aggregate Group Administrative State

By default, the dynamic aggregate group administrative state is enabled. The following subsections describe how to enable and disable a dynamic aggregate group’s administrative state with the **lacp linkagg admin state** command.

Enabling a Dynamic Aggregate Group

To enable the dynamic aggregate group administrative state, enter **lacp linkagg** followed by the dynamic aggregate group number and **admin state enable**. For example, to enable dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 admin state enable
```

Disabling a Dynamic Aggregate Group

To disable a dynamic aggregate group’s administrative state, use the **lacp linkagg admin state** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **admin state disable**.

For example, to disable dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 admin state disable
```

Configuring and Deleting the Dynamic Aggregate Group Actor Administrative Key

The following subsections describe how to configure and delete a dynamic aggregate group actor administrative key with the **lacp linkagg actor admin key** command.

Configuring a Dynamic Aggregate Actor Administrative Key

To configure the dynamic aggregate group actor switch administrative key enter **lacp linkagg** followed by the dynamic aggregate group number, **actor admin key**, and the value for the administrative key, which can be 0 through 65535.

For example, to configure dynamic aggregate group 4 with an administrative key of 10 you would enter:

```
-> lacp linkagg 4 actor admin key 10
```

Deleting a Dynamic Aggregate Actor Administrative Key

To remove an actor switch administrative key from a dynamic aggregate group's configuration use the **no** form of the **lacp linkagg actor admin key** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no actor admin key**.

For example, to remove an administrative key from dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 no actor admin key
```

Modifying the Dynamic Aggregate Group Actor System Priority

By default, the dynamic aggregate group actor system priority is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp linkagg actor system priority** command.

Configuring a Dynamic Aggregate Group Actor System Priority

You can configure a user-specified dynamic aggregate group actor system priority value to a value ranging from 0 to 65535 by entering **lacp linkagg** followed by the dynamic aggregate group number, **actor system priority**, and the new priority value.

For example, to change the actor system priority of dynamic aggregate group 4 to 2000 you would enter:

```
-> lacp linkagg 4 actor system priority 2000
```

Restoring the Dynamic Aggregate Group Actor System Priority

To restore the dynamic aggregate group actor system priority to its default (i.e., 0) value use the **no** form of the **lacp linkagg actor system priority** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no actor system priority**.

For example, to restore the actor system priority to its default value on dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 no actor system priority
```

Modifying the Dynamic Aggregate Group Actor System ID

By default, the dynamic aggregate group actor system ID (MAC address) is 00:00:00:00:00:00. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp linkagg actor system id** command.

Configuring a Dynamic Aggregate Group Actor System ID

You can configure a user-specified dynamic aggregate group actor system ID by entering **lacp linkagg** followed by the dynamic aggregate group number, **actor system id**, and the user-specified MAC address (in the hexadecimal format of *xx:xx:xx:xx:xx:xx*), which is used as the system ID.

For example, to configure the system ID on dynamic aggregate group 4 as 00:20:da:81:d5:b0 you would enter:

```
-> lacp linkagg 4 actor system id 00:20:da:81:d5:b0
```

Restoring the Dynamic Aggregate Group Actor System ID

To remove the user-configured actor switch system ID from a dynamic aggregate group's configuration use the **no** form of the **lacp linkagg actor system id** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no actor system id**.

For example, to remove the user-configured system ID from dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 no actor system id
```

Modifying the Dynamic Aggregate Group Partner Administrative Key

By default, the dynamic aggregate group partner administrative key (i.e., the administrative key of the partner switch) is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp linkagg partner admin key** command.

Configuring a Dynamic Aggregate Group Partner Administrative Key

You can modify the dynamic aggregate group partner administrative key to a value ranging from 0 to 65535 by entering **lacp linkagg** followed by the dynamic aggregate group number, **partner admin key**, and the value for the administrative key, which can be 0 through 65535.

For example, to set the partner administrative key to 4 on dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 partner admin key 10
```

Restoring the Dynamic Aggregate Group Partner Administrative Key

To remove a partner administrative key from a dynamic aggregate group's configuration use the **no** form of the **lacp linkagg partner admin key** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no partner admin key**.

For example, to remove the user-configured partner administrative key from dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 no partner admin key
```

Modifying the Dynamic Aggregate Group Partner System Priority

By default, the dynamic aggregate group partner system priority is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp linkagg partner system priority** command.

Configuring a Dynamic Aggregate Group Partner System Priority

You can modify the dynamic aggregate group partner system priority to a value ranging from 0 to 65535 by entering **lacp linkagg** followed by the dynamic aggregate group number, **partner system priority**, and the new priority value.

For example, to set the partner system priority on dynamic aggregate group 4 to 2000 you would enter:

```
-> lacp linkagg 4 partner system priority 2000
```

Restoring the Dynamic Aggregate Group Partner System Priority

To restore the dynamic aggregate group partner system priority to its default (i.e., 0) value use the **no** form of the **lacp linkagg partner system priority** command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no partner system priority**.

For example, to reset the partner system priority of dynamic aggregate group 4 to its default value you would enter:

```
-> lacp linkagg 4 no partner system priority
```

Modifying the Dynamic Aggregate Group Partner System ID

By default, the dynamic aggregate group partner system ID is 00:00:00:00:00:00. The following subsections describe how to configure a user-specified value and how to restore it to its default value with the [lacp linkagg partner system id](#) command.

Configuring a Dynamic Aggregate Group Partner System ID

You can configure the dynamic aggregate group partner system ID by entering **lacp linkagg** followed by the dynamic aggregate group number, **partner system id**, and the user-specified MAC address (in the hexadecimal format of *xx:xx:xx:xx:xx:xx*), which is used as the system ID.

For example, to configure the partner system ID as 00:20:da:81:d5:b0 on dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 partner system id 00:20:da:81:d5:b0
```

Restoring the Dynamic Aggregate Group Partner System ID

To remove the user-configured partner switch system ID from the dynamic aggregate group's configuration, use the **no** form of the [lacp linkagg partner system id](#) command by entering **lacp linkagg** followed by the dynamic aggregate group number and **no partner system id**.

For example, to remove the user-configured partner system ID from dynamic aggregate group 4 you would enter:

```
-> lacp linkagg 4 no partner system id
```

Modifying Dynamic Link Aggregate Actor Port Parameters

This section describes how to modify the following dynamic aggregate actor port parameters:

- Actor port administrative state (see [“Modifying the Actor Port System Administrative State” on page 24-19](#))
- Actor port system ID (see [“Modifying the Actor Port System ID” on page 24-20](#))
- Actor port system priority (see [“Modifying the Actor Port System Priority” on page 24-21](#))
- Actor port priority (see [“Modifying the Actor Port Priority” on page 24-22](#))

Note. See [“Configuring Ports to Join and Removing Ports in a Dynamic Aggregate Group” on page 24-12](#) for information on modifying a dynamic aggregate group administrative key.

All of the commands to modify actor port parameters allow you to add the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. However, these keywords do not modify a port's configuration. See [Chapter 1, “Configuring Ethernet Ports,”](#) for information on configuring Ethernet ports.

Note. A port may belong to only one aggregate group. In addition, mobile ports cannot be aggregated. See [Chapter 6, “Assigning Ports to VLANs,”](#) for more information on mobile ports.

Modifying the Actor Port System Administrative State

The system administrative state of a dynamic aggregate group actor port is indicated by bit settings in Link Aggregation Control Protocol Data Unit (LACPDU) frames sent by the port. By default, bits 0 (indicating that the port is active), 1 (indicating that short timeouts are used for LACPDU frames), and 2 (indicating that this port is available for aggregation) are set in LACPDU frames.

The following subsections describe how to configure user-specified values and how to restore them to their default values with the **lacp agg actor admin state** command.

Configuring Actor Port Administrative State Values

To configure an LACP actor port’s system administrative state values by entering **lacp agg**, the slot number, a slash (/), the port number, **actor admin state**, and one or more of the keywords shown in the table below *or none*:

lacp agg actor admin state Keyword	Definition
active	Specifies that bit 0 in LACPDU frames is set, which indicates that the link is able to exchange LACPDU frames. By default, this bit is set.
timeout	Specifies that bit 1 in LACPDU frames is set, which indicates that a short time-out is used for LACPDU frames. When this bit is disabled, a long time-out is used for LACPDU frames. By default, this bit is set.
aggregate	Specifies that bit 2 in LACPDU frames is set, which indicates that the system considers this link to be a potential candidate for aggregation. If this bit is not set, the system considers the link to be individual (it can only operate as a single link). By default, this bit is set.
synchronize	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 3) is set by the system, the port is allocated to the correct dynamic aggregation group. If this bit is not set by the system, the port is not allocated to the correct dynamic aggregation group.
collect	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 4) is set by the system, incoming LACPDU frames are collected from the individual ports that make up the dynamic aggregate group.
distribute	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 5) is set by the system, distributing outgoing frames on the port is disabled.
default	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 6) is set by the system, it indicates that the actor is using defaulted partner information administratively configured for the partner.

lACP agg actor admin state Keyword	Definition
expire	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 7) is set by the system, the actor cannot receive LACPDU frames.

Note. Specifying **none** removes all administrative states from the LACPDU configuration. For example:

```
-> lACP agg 5/49 actor admin state none
```

For example, to set bits 0 (**active**) and 2 (**aggregate**) on dynamic aggregate actor port 49 in slot 5 you would enter:

```
-> lACP agg 5/49 actor admin state active aggregate
```

As an option you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to set bits 0 (**active**) and 2 (**aggregate**) on dynamic aggregate actor port 49 in slot 5 and document that the port is a Gigabit Ethernet port you would enter:

```
-> lACP agg gigaethernet 5/49 actor admin state active aggregate
```

Restoring Actor Port Administrative State Values

To restore LACPDU bit settings to their default values, use the **lACP agg actor admin state** command by entering **no** before the **active**, **timeout**, and **aggregate** keywords.

For example, to restore bits 0 (**active**) and 2 (**aggregate**) to their default settings on dynamic aggregate actor port 2 in slot 5 you would enter:

```
-> lACP agg 5/2 actor admin state no active no aggregate
```

Note. Since individual bits with the LACPDU frame are set with the **lACP agg actor admin state** command you can set some bits on and restore other bits within the same command. For example, if you wanted to restore bit 2 (**aggregate**) to its default settings and set bit 0 (**active**) on dynamic aggregate actor port 49 in slot 5 you would enter:

```
-> lACP agg 5/49 actor admin state active no aggregate
```

Modifying the Actor Port System ID

By default, the actor port system ID (i.e., the MAC address used as the system ID on dynamic aggregate actor ports) is 00:00:00:00:00:00. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lACP agg actor system id** command.

Configuring an Actor Port System ID

You can configure the actor port system ID by entering **lACP agg**, the slot number, a slash (/), the port number, **actor system id**, and the user specified actor port system ID (i.e., MAC address) in the hexadecimal format of xx:xx:xx:xx:xx:xx.

For example, to modify the system ID of the dynamic aggregate actor port 3 in slot 7 to **00:20:da:06:ba:d3** you would enter:

```
-> lacp agg 7/3 actor system id 00:20:da:06:ba:d3
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the system ID of the dynamic aggregate actor port 3 in slot 7 to **00:20:da:06:ba:d3** and document that the port is 10 Mbps Ethernet you would enter:

```
-> lacp agg ethernet 7/3 actor system id 00:20:da:06:ba:d3
```

Restoring the Actor Port System ID

To remove a user-configured system ID from a dynamic aggregate group actor port's configuration use the **no** form of the **lacp agg actor system id** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no actor system id**.

For example, to remove a user-configured system ID from dynamic aggregate actor port 3 in slot 7 you would enter:

```
-> lacp agg 7/3 no actor system id
```

Modifying the Actor Port System Priority

By default, the actor system priority is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp agg actor system priority** command.

Configuring an Actor Port System Priority

You can configure the actor system priority to a value ranging from 0 to 255 by entering **lacp agg**, the slot number, a slash (/), the port number, **actor system priority**, and the user-specified actor port system priority.

For example, to modify the system priority of dynamic aggregate actor port 5 in slot 2 to 200 you would enter:

```
-> lacp agg 2/5 actor system priority 200
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the system priority of dynamic aggregate actor port 5 in slot 2 to 200 and document that the port is a Giga Ethernet port you would enter:

```
-> lacp agg gigaethernet 2/5 actor system priority 200
```

Restoring the Actor Port System Priority

To remove a user-configured actor port system priority from a dynamic aggregate group actor port's configuration use the **no** form of the **lacp agg actor system priority** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no actor system priority**.

For example, to remove a user-configured system priority from dynamic aggregate actor port 5 in slot 2 you would enter:

```
-> lacp agg 2/5 no actor system priority
```


Modifying the Actor Port Priority

By default, the actor port priority (used to converge dynamic key changes) is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp agg actor port priority** command.

Configuring the Actor Port Priority

You can configure the actor port priority to a value ranging from 0 to 255 by entering **lacp agg**, the slot number, a slash (/), the port number, **actor port priority**, and the user-specified actor port priority.

For example, to modify the actor port priority of dynamic aggregate actor port 1 in slot 2 to 100 you would enter:

```
-> lacp agg 2/1 actor port priority 100
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the actor port priority of dynamic aggregate actor port 1 in slot 2 to 100 and document that the port is a Giga Ethernet port you would enter:

```
-> lacp agg gigaethernet 2/1 actor port priority 100
```

Restoring the Actor Port Priority

To remove a user configured actor port priority from a dynamic aggregate group actor port's configuration use the **no** form of the **lacp agg actor port priority** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no actor port priority**.

For example, to remove a user-configured actor priority from dynamic aggregate actor port 1 in slot 2 you would enter:

```
-> lacp agg 2/1 no actor port priority
```


Modifying Dynamic Aggregate Partner Port Parameters

This section describes how to modify the following dynamic aggregate partner port parameters:

- Partner port system administrative state (see “[Modifying the Partner Port System Administrative State](#)” on page 24-23)
- Partner port administrative key (see “[Modifying the Partner Port Administrative Key](#)” on page 24-25)
- Partner port system ID (see “[Modifying the Partner Port System ID](#)” on page 24-25)
- Partner port system priority (see “[Modifying the Partner Port System Priority](#)” on page 24-26)
- Partner port administrative state (see “[Modifying the Partner Port Administrative Status](#)” on page 24-27)
- Partner port priority (see “[Modifying the Partner Port Priority](#)” on page 24-27)

All of the commands to modify partner port parameters allow you to add the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. However, these keywords do not modify a port’s configuration. See [Chapter 1, “Configuring Ethernet Ports,”](#) for information on configuring Ethernet ports.

Note. A port may belong to only one aggregate group. In addition, mobile ports cannot be aggregated. See [Chapter 6, “Assigning Ports to VLANs,”](#) for more information on mobile ports.

Modifying the Partner Port System Administrative State

The system administrative state of a dynamic aggregate group partner (i.e., remote switch) port is indicated by bit settings in Link Aggregation Control Protocol Data Unit (LACPDU) frames sent by this port. By default, bits 0 (indicating that the port is active), 1 (indicating that short timeouts are used for LACPDU frames), and 2 (indicating that this port is available for aggregation) are set in LACPDU frames.

The following subsections describe how to configure user-specified values and how to restore them to their default values with the **lacp agg partner admin state** command.

Configuring Partner Port System Administrative State Values

To configure the dynamic aggregate partner port’s system administrative state values by entering **lacp agg**, the slot number, a slash (/), the port number, **partner admin state**, and one or more of the keywords shown in the table below *or none*:

Keyword	Definition
active	Specifies that bit 0 in LACPDU frames is set, which indicates that the link is able to exchange LACPDU frames. By default, this bit is set.
timeout	Specifies that bit 1 in LACPDU frames is set, which indicates that a short time-out is used for LACPDU frames. When this bit is disabled, a long time-out is used for LACPDU frames. By default, this bit is set.
aggregate	Specifies that bit 2 in LACPDU frames is set, which indicates that the system considers this link to be a potential candidate for aggregation. If this bit is not set, the system considers the link to be individual (it can only operate as a single link). By default, this bit is set.

Keyword	Definition
synchronize	Specifies that bit 3 in the partner state octet is enabled. When this bit is set, the port is allocated to the correct dynamic aggregation group. If this bit is not enabled, the port is not allocated to the correct aggregation group. By default, this value is disabled.
collect	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 4) is set by the system, incoming LACPDU frames are collected from the individual ports that make up the dynamic aggregate group.
distribute	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 5) is set by the system, distributing outgoing frames on the port is disabled.
default	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 6) is set by the system, it indicates that the partner is using defaulted actor information administratively configured for the partner.
expire	Specifying this keyword has no effect because the system always determines its value. When this bit (bit 7) is set by the system, the actor cannot receive LACPDU frames.

Note. Specifying **none** removes all administrative states from the LACPDU configuration. For example:

```
-> lacp agg 7/49 partner admin state none
```

For example, to set bits 0 (**active**) and 2 (**aggregate**) on dynamic aggregate partner port 49 in slot 7 you would enter:

```
-> lacp agg 7/49 partner admin state active aggregate
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to set bits 0 (**active**) and 2 (**aggregate**) on dynamic aggregate partner port 49 in slot 7 and document that the port is a Gigabit Ethernet port you would enter:

```
-> lacp agg gigaethernet 7/49 partner admin state active aggregate
```

Restoring Partner Port System Administrative State Values

To restore LACPDU bit settings to their default values use the **no** form of the **lacp agg partner admin state** command by entering **no** before the **active**, **timeout**, **aggregate**, or **synchronize** keywords.

For example, to restore bits 0 (**active**) and 2 (**aggregate**) to their default settings on dynamic aggregate partner port 1 in slot 7 you would enter:

```
-> lacp agg 7/1 partner admin state no active no aggregate
```

Note. Since individual bits with the LACPDU frame are set with the **lacp agg partner admin state** command you can set some bits on and restore other bits to default values within the same command. For example, if you wanted to restore bit 2 (**aggregate**) to its default settings and set bit 0 (**active**) on dynamic aggregate partner port 1 in slot 7 you would enter:

```
-> lacp agg 7/1 partner admin state active no aggregate
```

Modifying the Partner Port Administrative Key

By default, the dynamic aggregate partner port's administrative key is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp agg partner admin key** command.

Configuring the Partner Port Administrative Key

You can configure the dynamic aggregate partner port's administrative key to a value ranging from 0 to 65535 by entering **lacp agg**, the slot number, a slash (/), the port number, **partner admin key**, and the user-specified partner port administrative key.

For example, to modify the administrative key of a dynamic aggregate group partner port 1 in slot 6 to 1000 enter:

```
-> lacp agg 6/1 partner admin key 1000
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the administrative key of a dynamic aggregate group partner port 1 in slot 6 to 1000 and document that the port is a 10 Mbps Ethernet port you would enter:

```
-> lacp agg ethernet 6/1 partner admin key 1000
```

Restoring the Partner Port Administrative Key

To remove a user-configured administrative key from a dynamic aggregate group partner port's configuration use the **no** form of the **lacp agg partner admin key** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no partner admin key**.

For example, to remove the user-configured administrative key from dynamic aggregate partner port 1 in slot 6, enter:

```
-> lacp agg 6/1 no partner admin key
```

Modifying the Partner Port System ID

By default, the partner port system ID (i.e., the MAC address used as the system ID on dynamic aggregate partner ports) is 00:00:00:00:00:00. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp agg partner admin system id** command.

Configuring the Partner Port System ID

You can configure the partner port system ID by entering **lACP agg**, the slot number, a slash (/), the port number, **partner admin system id**, and the user-specified partner administrative system ID (i.e., the MAC address in hexadecimal format).

For example, to modify the system ID of dynamic aggregate partner port 49 in slot 6 to **00:20:da:06:ba:d3** you would enter:

```
-> lACP agg 6/49 partner admin system id 00:20:da:06:ba:d3
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the system ID of dynamic aggregate partner port 49 in slot 6 to **00:20:da:06:ba:d3** and document that the port is a Gigabit Ethernet port you would enter:

```
-> lACP agg gigaethernet 6/49 partner admin system id 00:20:da:06:ba:d3
```

Restoring the Partner Port System ID

To remove a user-configured system ID from a dynamic aggregate group partner port's configuration use the **no** form of the **lACP agg partner admin system id** command by entering **lACP agg**, the slot number, a slash (/), the port number, and **no partner admin system id**.

For example, to remove a user-configured system ID from dynamic aggregate partner port 2 in slot 6 you would enter:

```
-> lACP agg 6/2 no partner admin system id
```

Modifying the Partner Port System Priority

By default, the administrative priority of a dynamic aggregate group partner port is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lACP agg partner admin system priority** command.

Configuring the Partner Port System Priority

You can configure the administrative priority of a dynamic aggregate group partner port to a value ranging from 0 to 255 by entering **lACP agg**, the slot number, a slash (/), the port number, **partner admin system priority**, and the user-specified administrative system priority.

For example, to modify the administrative priority of a dynamic aggregate partner port 49 in slot 4 to 100 you would enter:

```
-> lACP agg 4/49 partner admin system priority 100
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the administrative priority of dynamic aggregate partner port 49 in slot 4 to 100 and specify that the port is a Gigabit Ethernet port you would enter:

```
-> lACP agg gigaethernet 4/49 partner admin system priority 100
```

Restoring the Partner Port System Priority

To remove a user-configured system priority from a dynamic aggregate group partner port's configuration use the **no** form of the **lacp agg partner admin system priority** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no partner admin system priority**.

For example, to remove a user-configured system ID from dynamic aggregate partner port 3 in slot 4 you would enter:

```
-> lacp agg 4/3 no partner admin system priority
```

Modifying the Partner Port Administrative Status

By default, the administrative status of a dynamic aggregate group partner port is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp agg partner admin port** command.

Configuring the Partner Port Administrative Status

You can configure the administrative status of a dynamic aggregate group partner port to a value ranging from 0 to 65535 by entering **lacp agg**, the slot number, a slash (/), the port number, **partner admin port**, and the user-specified partner port administrative status.

For example, to modify the administrative status of dynamic aggregate partner port 1 in slot 7 to 200 you would enter:

```
-> lacp agg 7/1 partner admin port 200
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the administrative status of dynamic aggregate partner port 1 in slot 7 to 200 and document that the port is a Giga Ethernet port you would enter:

```
-> lacp agg gigaethernet 7/1 partner admin port 200
```

Restoring the Partner Port Administrative Status

To remove a user-configured administrative status from a dynamic aggregate group partner port's configuration use the **no** form of the **lacp agg partner admin port** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no partner admin port**.

For example, to remove a user-configured administrative status from dynamic aggregate partner port 1 in slot 7 you would enter:

```
-> lacp agg 7/1 no partner admin port
```

Modifying the Partner Port Priority

The default partner port priority is 0. The following subsections describe how to configure a user-specified value and how to restore the value to its default value with the **lacp agg partner admin port priority** command.

Configuring the Partner Port Priority

To configure the partner port priority to a value ranging from 0 to 255 by entering **lacp agg**, the slot number, a slash (/), the port number, **partner admin port priority**, and the user-specified partner port priority.

For example, to modify the port priority of dynamic aggregate partner port 3 in slot 4 to 100 you would enter:

```
-> lacp agg 4/3 partner admin port priority 100
```

As an option, you can use the **ethernet**, **fastethernet**, and **gigaethernet** keywords before the slot and port number to document the interface type or make the command look consistent with early-generation Alcatel-Lucent CLI syntax. For example, to modify the port priority of dynamic aggregate partner port 3 in slot 4 to 100 and document that the port is a Giga Ethernet port you would enter:

```
-> lacp agg gigaethernet 4/3 partner admin port priority 100
```

Restoring the Partner Port Priority

To remove a user-configured partner port priority from a dynamic aggregate group partner port's configuration use the **no** form of the **lacp agg partner admin port priority** command by entering **lacp agg**, the slot number, a slash (/), the port number, and **no partner admin port priority**.

For example, to remove a user-configured partner port priority from dynamic aggregate partner port 3 in slot 4 you would enter:

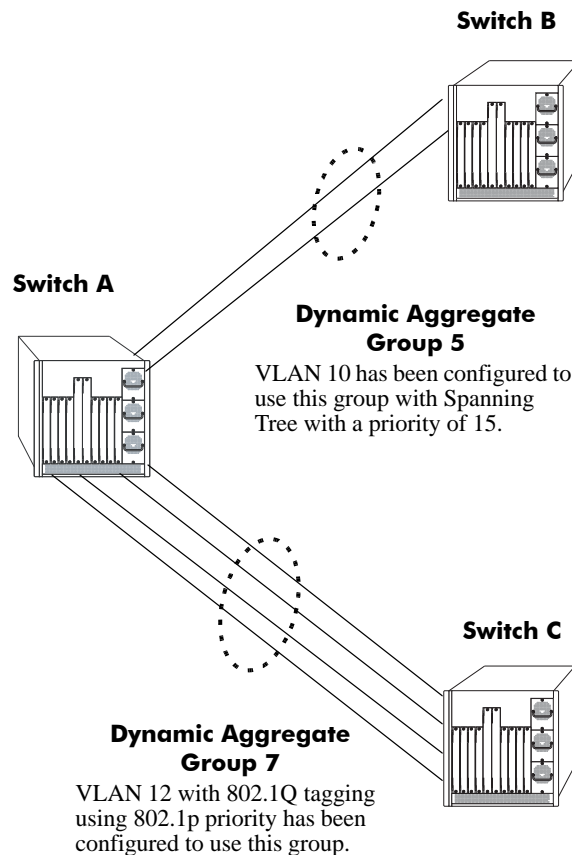
```
-> lacp agg 4/3 no partner admin port priority
```

Application Examples

Dynamic link aggregation groups are treated by the switch's software the same way it treats individual physical ports. This section demonstrates this feature by providing sample network configurations that use dynamic aggregation along with other software features. In addition, tutorials are provided that show how to configure these sample networks by using Command Line Interface (CLI) commands.

Sample Network Overview

The figure below shows two VLANs on Switch A that use two different link aggregation groups. VLAN 10 has been configured on dynamic aggregate group 5 with Spanning Tree Protocol (STP) with the highest (15) priority possible. And VLAN 12 has been configured on dynamic aggregate group 7 with 802.1Q tagging and 802.1p priority bit settings.



Sample Network Using Dynamic Link Aggregation

The steps to configure VLAN 10 (Spanning Tree example) are described in [“Link Aggregation and Spanning Tree Example”](#) on page 24-30. The steps to configure VLAN 12 (802.1Q and 802.1p example) are described in [“Link Aggregation and QoS Example”](#) on page 24-31.

Note. Although you would need to configure both the local (i.e., Switch A) and remote (i.e., Switches B and C) switches, only the steps to configure the local switch are provided since the steps to configure the remote switches are not significantly different.

Link Aggregation and QoS Example

As shown in the figure on [page 24-29](#), VLAN 12, which uses 802.1Q frame tagging and 802.1p prioritization, has been configured to use dynamic aggregate group 7. The actual physical links connect ports 4/1, 4/2, 4/3, and 4/4 on Switch A to ports 1/1, 1/2, 1/3, and 1/4 on Switch C. Follow the steps below to configure this network:

Note. Only the steps to configure the local (i.e., Switch A) switch are provided here since the steps to configure the remote (i.e., Switch C) switch would not be significantly different.

- 1 Configure dynamic aggregate group 7 by entering:

```
-> lacp linkagg 7 size 4
```

- 2 Configure ports 4/1, 4/2, 4/3, and 4/4 the same actor administrative key (7) by entering:

```
-> lacp agg 4/1 actor admin key 7
-> lacp agg 4/2 actor admin key 7
-> lacp agg 4/3 actor admin key 7
-> lacp agg 4/4 actor admin key 7
```

- 3 Create VLAN 12 by entering:

```
-> vlan 12
```

- 4 Configure 802.1Q tagging with a tagging ID (i.e., VLAN ID) of 12 on dynamic aggregate group 7 by entering:

```
-> vlan 12 802.1q 7
```

- 5 If the QoS Manager has been disabled (it is enabled by default) enable it by entering:

```
-> qos enable
```

Note. *Optional.* Use the [show qos config](#) command to determine if the QoS Manager is enabled or disabled.

- 6 Configure a policy condition for VLAN 12 called “vlan12_condition” by entering:

```
-> policy condition vlan12_condition destination vlan 12
```

- 7 Configure an 802.1p policy action with the highest priority possible (i.e., 7) for VLAN 12 called “vlan12_action” by entering:

```
-> policy action vlan12_action 802.1p 7
```

- 8 Configure a QoS rule called “vlan12_rule” by using the policy condition and policy rules you configured in steps 8 and 9 above by entering:

```
-> policy rule vlan12_rule enable condition vlan12_condition action
vlan12_action
```

- 9 Enable your 802.1p QoS settings by entering **qos apply** as shown below:

```
-> qos apply
```


A screen similar to the following would be displayed:

```
Dynamic Aggregable Port
  SNMP Id                : 2001,
  Slot/Port              : 2/1,
  Administrative State   : ENABLED,
  Operational State      : DOWN,
  Port State              : CONFIGURED,
  Link State              : DOWN,
  Selected Agg Number    : NONE,
  Primary port           : UNKNOWN,
LACP
  Actor System Priority   : 10,
  Actor System Id        : [00:d0:95:6a:78:3a],
  Actor Admin Key        : 8,
  Actor Oper Key         : 8,
  Partner Admin System Priority : 20,
  Partner Oper System Priority : 20,
  Partner Admin System Id : [00:00:00:00:00:00],
  Partner Oper System Id  : [00:00:00:00:00:00],
  Partner Admin Key      : 8,
  Partner Oper Key       : 0,
  Attached Agg Id        : 0,
  Actor Port             : 7,
  Actor Port Priority     : 15,
  Partner Admin Port     : 0,
  Partner Oper Port      : 0,
  Partner Admin Port Priority : 0,
  Partner Oper Port Priority : 0,
  Actor Admin State      : act1.tim1.aggl.syn0.col0.dis0.def1.exp0,
  Actor Oper State       : act1.tim1.aggl.syn0.col0.dis0.def1.exp0,
  Partner Admin State    : act0.tim0.aggl.syn1.col1.dis1.def1.exp0,
  Partner Oper State     : act0.tim0.aggl.syn0.col1.dis1.def1.exp0
```

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of **show** commands for link aggregation.

25 Configuring IP

Internet Protocol (IP) is primarily a network-layer (Layer 3) protocol that contains addressing and control information that enables packets to be forwarded. Along with Transmission Control Protocol (TCP), IP represents the heart of the Internet protocols. IP has two primary responsibilities, providing connectionless, best-effort delivery of datagrams through an internetwork; and providing fragmentation and reassembly of datagrams to support data links with different Maximum Transmission Unit (MTU) sizes.

Note. IP routing (Layer 3) can be accomplished using static routes or by using an IP routing protocol such as Routing Information Protocol (RIP) For more information see [Chapter 23, “Configuring RIP”](#).

There are two versions of Internet Protocol supported, IPv4 and IPv6. For more information about using IPv6, see [Chapter 26, “Configuring IPv6.”](#)

In This Chapter

This chapter describes IP and how to configure it through the Command Line Interface (CLI). It includes instructions for enabling IP forwarding, configuring IP route maps, as well as basic IP configuration commands (e.g., `ip default-ttl`). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*. This chapter provides an overview of IP and includes information about the following procedures:

- IP Forwarding
 - Configuring an IP Router Interface (see [page 25-8](#))
 - Creating a Static Route (see [page 25-10](#))
 - Creating a Default Route (see [page 25-11](#))
 - Configuring Address Resolution Protocol (ARP) (see [page 25-12](#))
- IP Configuration
 - Configuring a DHCP Client Interface (see [page 25-15](#))
 - Configuring the Router Primary Address (see [page 25-15](#))
 - Configuring the Router ID (see [page 25-15](#))
 - Configuring the Time-to-Live (TTL) Value (see [page 25-16](#))
 - Configuring Route Map Redistribution (see [page 25-16](#))
 - IP-Directed Broadcasts (see [page 25-23](#))
 - Protecting the Switch from Denial of Service (DoS) attacks (see [page 25-23](#))

- Managing IP
 - Internet Control Message Protocol (ICMP) (see [page 25-29](#))
 - Using the Ping Command (see [page 25-32](#))
 - Tracing an IP Route (see [page 25-33](#))
 - Displaying TCP Information (see [page 25-33](#))
 - Displaying User Datagram Protocol (UDP) Information (see [page 25-33](#))

IP Specifications

Note that the maximum limit values provided in the following Specifications table are subject to available system resources:

RFCs Supported	RFC 791–Internet Protocol RFC 792–Internet Control Message Protocol RFC 826–An Ethernet Address Resolution Protocol 2784– <i>Generic Routing Encapsulation (GRE)</i> 2890– <i>Key and Sequence Number Extensions to GRE</i> (extensions defined are not supported) 1701– <i>Generic Routing Encapsulation (GRE)</i> 1702– <i>Generic Routing Encapsulation over IPV4 Networks</i> 2003–IP Encapsulation within IP.
Platforms Supported	OmniSwitch 6250 Series
Maximum VLANs per switch	4094
Maximum router IP interfaces per switch	128
Maximum IP router interfaces per VLAN	8
Maximum ARP entries per switch	256
Maximum ARP filters per switch	200
Maximum IP static routes per switch	256
Maximum IP host routes per switch	256

IP Defaults

The following table lists the defaults for IP configuration through the **ip** command.

Description	Command	Default
IP-Directed Broadcasts	ip directed-broadcast	off
Time-to-Live Value	ip default-ttl	64 (hops)
IP interfaces	ip interface	VLAN 1 interface.
ARP filters	ip dos arp-poison restricted-address	0

Quick Steps for Configuring IP Forwarding

Using only IP, which is always enabled on the switch, devices connected to ports on the same VLAN are able to communicate at Layer 2. The initial configuration for all Alcatel-Lucent switches consists of a default VLAN 1. All switch ports are initially assigned to this VLAN. In addition, when a stackable OmniSwitch is added to a stack of switches or a switching module is added to a chassis-based OmniSwitch, all ports belonging to the new switch and/or module are also assigned to VLAN 1. If additional VLANs are not configured on the switch, the entire switch is treated as one large broadcast domain, and all ports receive all traffic from all other ports.

Note. The operational status of a VLAN remains inactive until at least one active switch port is assigned to the VLAN. Ports are considered active if they are connected to an active network device. Non-active port assignments are allowed, but do not change the operational state of the VLAN.

To forward packets to a different VLAN on a switch, you must create a router interface on each VLAN. The following steps show you how to enable IP forwarding between VLANs “from scratch”. If active VLANs have already been created on the switch, you only need to create router interfaces on each VLAN (Steps 5 and 6).

- 1 Create VLAN 1 with a description (e.g., VLAN 1) by using the **vlan** command. For example:

```
-> vlan 1 name "VLAN 1"
```

- 2 Create VLAN 2 with a description (e.g., VLAN 2) by using the **vlan** command. For example:

```
-> vlan 2 name "VLAN 2"
```

- 3 Assign an active port to VLAN 1 by using the **vlan port default** command. For example, the following command assigns port 1 on slot 1 to VLAN 1:

```
-> vlan 1 port default 1/1
```

- 4 Assign an active port to VLAN 2 by using the **vlan port default** command. For example, the following command assigns port 2 on slot 1 to VLAN 2:

```
-> vlan 2 port default 1/2
```

- 5 Create an IP router interface on VLAN 1 using the **ip interface** command. For example:

```
-> ip interface vlan-1 address 171.10.1.1 vlan 1
```

- 6 Create an IP router interface on VLAN 2 using the **ip interface** command. For example:

```
-> ip interface vlan-2 address 171.11.1.1 vlan 2
```

Note. See [Chapter 4, “Configuring VLANs.”](#) for more information about how to create VLANs and VLAN router interfaces.

IP Overview

IP is a network-layer (Layer 3) protocol that contains addressing and control information that enables packets to be forwarded on a network. IP is the primary network-layer protocol in the Internet protocol suite. Along with TCP, IP represents the heart of the Internet protocols.

IP Protocols

IP is associated with several Layer 3 and Layer 4 protocols. These protocols are built into the base code loaded on the switch. A brief overview of supported IP protocols is included below.

Transport Protocols

IP is both connectionless (it forwards each datagram separately) and unreliable (it does not guarantee delivery of datagrams). This means that a datagram may be damaged in transit, thrown away by a busy switch, or simply never make it to its destination. The resolution of these transit problems is to use a Layer 4 transport protocol, such as:

- TCP—A major data transport mechanism that provides reliable, connection-oriented, full-duplex data streams. While the role of TCP is to add reliability to IP, TCP relies upon IP to do the actual delivering of datagrams.
- UDP—A secondary transport-layer protocol that uses IP for delivery. UDP is not connection-oriented and does not provide reliable end-to-end delivery of datagrams. But some applications can safely use UDP to send datagrams that do not require the extra overhead added by TCP. For more information on UDP, see [Chapter 29, “Configuring DHCP.”](#)

Application-Layer Protocols

Application-layer protocols are used for switch configuration and management:

- Bootstrap Protocol (BOOTP)/Dynamic Host Configuration Protocol (DHCP)—May be used by an end station to obtain an IP address. The switch provides a DHCP Relay that allows BOOTP requests/replies to cross different networks.
- Simple Network Management Protocol (SNMP)—Allows communication between SNMP managers and SNMP agents on an IP network. Network administrators use SNMP to monitor network performance and manage network resources. For more information, see the “Using SNMP” chapter in the *OmniSwitch 6250 Switch Management Guide*.
- Telnet—Used for remote connections to a device. You can telnet to a switch and configure the switch and the network by using the CLI.
- File Transfer Protocol (FTP)—Enables the transfer of files between hosts. This protocol is used to load new images onto the switch.

Additional IP Protocols

There are several additional IP-related protocols that may be used with IP forwarding. These protocols are included as part of the base code.

- Address Resolution Protocol (ARP)—Used to match the IP address of a device with its physical (MAC) address. For more information, see [“Configuring Address Resolution Protocol \(ARP\)” on page 25-12.](#)
- Internet Control Message Protocol (ICMP)—Specifies the generation of error messages, test packets, and informational messages related to IP. ICMP supports the **ping** command used to determine if hosts are online. For more information, see [“Internet Control Message Protocol \(ICMP\)” on page 25-29.](#)
- Router Discovery Protocol (RDP)—Used to advertise and discover routers on the LAN. For more information, see [Chapter 28, “Configuring RDP.”](#)
- Multicast Services—Includes IP multicast switching (IPMS). For more information, see [Chapter 36, “Configuring IP Multicast Switching.”](#)

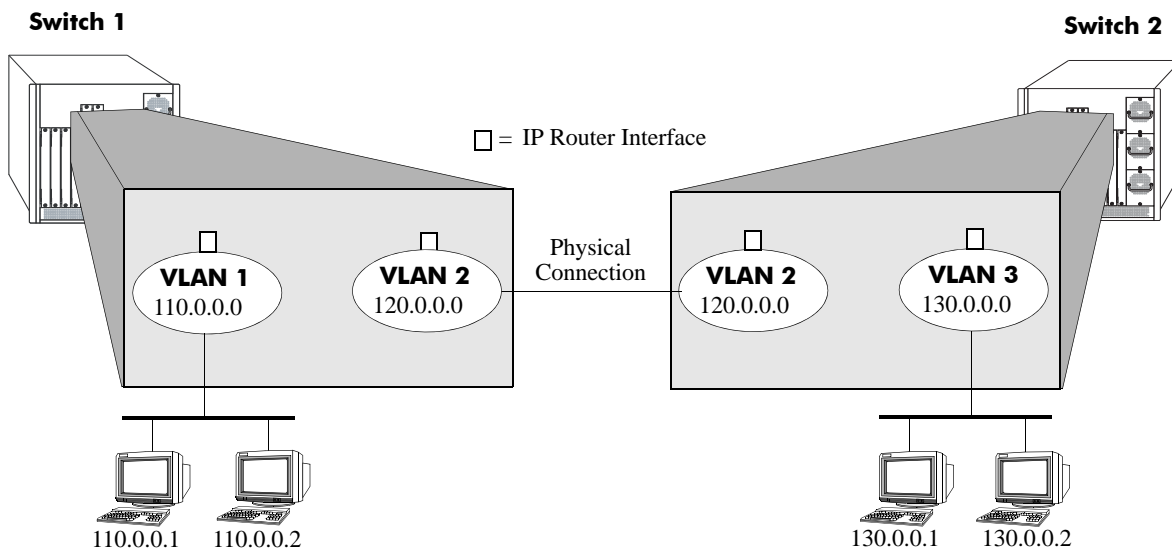
IP Forwarding

Network device traffic is bridged (switched) at the Layer 2 level between ports that are assigned to the same VLAN. However, if a device needs to communicate with another device that belongs to a different VLAN, then Layer 3 routing is necessary to transmit traffic between the VLANs. Bridging makes the decision on where to forward packets based on the packet's destination MAC address; routing makes the decision on where to forward packets based on the packet's IP network address (e.g., IP - 21.0.0.10).

Alcatel-Lucent switches support routing of IP traffic. A VLAN is available for routing when at least one router interface is defined for that VLAN and at least one active port is associated with the VLAN. If a VLAN does not have a router interface, the ports associated with that VLAN are in essence firewalled from other VLANs.

IP multinetting is also supported. A network is said to be multinetted when multiple IP subnets are brought together within a single broadcast domain. It is now possible to configure up to eight IP interfaces per VLAN. Each interface is configured with a different subnet. As a result, traffic from each configured subnet can coexist on the same VLAN.

In the illustration below, an IP router interface has been configured on each VLAN. Therefore, workstations connected to ports on VLAN 1 on Switch 1 can communicate with VLAN 2; and workstations connected to ports on VLAN 3 on Switch 2 can communicate with VLAN 2. Also, ports from both switches have been assigned to VLAN 2, and a physical connection has been made between the switches. Therefore, workstations connected to VLAN 1 on Switch 1 can communicate with workstations connected to VLAN 3 on Switch 2.



IP Forwarding

If the switch is running in single MAC router mode, a maximum of 4094 VLANs can have IP interfaces defined. In this mode, each router VLAN is assigned the same MAC address, which is the base chassis MAC address for the switch.

Configuring an IP Router Interface

IP is enabled by default. Using IP, devices connected to ports on the same VLAN are able to communicate. However, to forward packets to a different VLAN, you must create at least one router interface on each VLAN.

Use the **ip interface** command to define up to eight IP interfaces for an existing VLAN. The following parameter values are configured with this command:

- A unique interface name (text string up to 20 characters) is used to identify the IP interface. Specifying this parameter is required to create or modify an IP interface.
- The VLAN ID of an existing VLAN.
- An IP address to assign to the router interface (e.g., 193.204.173.21). Note that router interface IP addresses must be unique. You cannot have two router interfaces with the same IP address.
- A subnet mask (defaults to the IP address class). It is possible to specify the mask in dotted decimal notation (e.g., 255.255.0.0) or with a slash (/) after the IP address followed by the number of bits to specify the mask length (e.g., 193.204.173.21/64).
- The forwarding status for the interface (defaults to forwarding). A forwarding router interface sends IP frames to other subnets. A router interface that is not forwarding can receive frames from other hosts on the same subnet.
- An Ethernet-II or SNAP encapsulation for the interface (defaults to Ethernet-II). The encapsulation determines the framing type the interface uses when generating frames that are forwarded out of VLAN ports. Select an encapsulation that matches the encapsulation of the majority of VLAN traffic.
- The Local Proxy ARP status for the VLAN. If enabled, traffic within the VLAN is routed instead of bridged. ARP requests return the MAC address of the IP router interface defined for the VLAN. For more information about Local Proxy ARP, see [“Local Proxy ARP” on page 25-13](#).
- The primary interface status. Designates the specified IP interface as the primary interface for the VLAN. By default, the first interface bound to a VLAN becomes the primary interface for that VLAN.

The following **ip interface** command example creates an IP interface named Marketing with an IP network address of 21.0.0.1 and binds the interface to VLAN 455:

```
-> ip interface Marketing address 21.0.0.1 vlan 455
```

The **name** parameter is the only parameter required with this command. Specifying additional parameters is only necessary to configure a value other than the default value for that parameter. For example, all of the following commands will create an IP router interface for VLAN 955 with a class A subnet mask, an enabled forwarding status, Ethernet-II encapsulation, and a disabled Local Proxy ARP and primary interface status:

```
-> ip interface Accounting address 71.0.0.1 mask 255.0.0.0 vlan 955 forward e2  
no local-proxy-arp no primary  
-> ip interface Accounting address 71.0.0.1/8 vlan 955  
-> ip interface Accounting address 71.0.0.1 vlan 955
```

Modifying an IP Router Interface

The **ip interface** command is also used to modify existing IP interface parameter values. It is not necessary to first remove the IP interface and then create it again with the new values. The changes specified will overwrite existing parameter values. For example, the following command changes the subnet mask to **255.255.255.0**, the forwarding status to **no forwarding** and the encapsulation to **snap** by overwriting existing parameter values defined for the interface. The interface name, **Accounting**, is specified as part of the command syntax to identify which interface to change.

```
-> ip interface Accounting mask 255.255.255.0 no forward snap
```

Note that when changing the IP address for the interface, the subnet mask will revert back to the default mask value if it was previously set to a non-default value and it is not specified when changing the IP address. For example, the following command changes the IP address for the Accounting interface:

```
-> ip interface Accounting address 40.0.0.1
```

The subnet mask for the Accounting interface was previously set to 255.255.255.0. The above example resets the mask to the default value of 255.0.0.0 because 40.0.0.1 is a Class A address and no other mask was specified with the command. This only occurs when the IP address is modified; all other parameter values remain unchanged unless otherwise specified.

To avoid the problem in the above example, simply enter the non-default mask value whenever the IP address is changed for the interface. For example:

```
-> ip interface Accounting address 40.0.0.1 mask 255.255.255.0  
-> ip interface Accounting address 40.0.0.1/8
```

Use the **show ip interface** command to verify IP router interface changes. For more information about these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Removing an IP Router Interface

To remove an IP router interface, use the **no** form of the **ip interface** command. Note that it is only necessary to specify the name of the IP interface, as shown in the following example:

```
-> no ip interface Marketing
```

To view a list of IP interfaces configured on the switch, use the **show ip interface** command. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuring a Loopback0 Interface

Loopback0 is the name assigned to an IP interface to identify a consistent address for network management purposes. The Loopback0 interface is not bound to any VLAN, so it will always remain operationally active. This differs from other IP interfaces in that if there are no active ports in the VLAN, all IP interface associated with that VLAN are not active. In addition, the Loopback0 interface provides a unique IP address for the switch that is easily identifiable to network management applications.

This type of interface is created in the same manner as all other IP interfaces, using the [ip interface](#) command. To identify a Loopback0 interface, enter **Loopback0** for the interface name. For example, the following command creates the Loopback0 interface with an IP address of 10.11.4.1:

```
-> ip interface Loopback0 address 10.11.4.1
```

Note the following when configuring the Loopback0 interface:

- The interface name, “Loopback0”, is case sensitive.
- The **admin** parameter is the only configurable parameter supported with this type of interface.
- The Loopback0 interface is always active and available.
- Only one Loopback0 interface per switch is allowed.
- Creating this interface does *not* deduct from the total number of IP interfaces allowed per VLAN or switch.

Loopback0 Address Advertisement

The Loopback0 IP interface address is automatically advertised by the IGP protocol RIP when the interface is created. There is no additional configuration necessary to trigger advertisement with this protocol.

Note that RIP advertises the host route to the Loopback0 IP interface as a redistributed (directhost) route..

Creating a Static Route

Static routes are user-defined and carry a higher priority than routes created by dynamic routing protocols. That is, if two routes have the same metric value, the static route has the higher priority. Static routes allow you to define, or customize, an explicit path to an IP network segment, which is then added to the IP Forwarding table. Static routes can be created between VLANs to enable devices on these VLANs to communicate.

Use the [ip static-route](#) command to create a static route. You must specify the destination IP address of the route as well as the IP address of the first hop (gateway) used to reach the destination. For example, to create a static route to IP address 171.11.0.0 through gateway 171.11.2.1, you would enter:

```
-> ip static-route 171.11.0.0 gateway 171.11.2.1
```

The subnet mask is not required if you want to use the natural subnet mask. By default, the switch imposes a natural mask on the IP address. In the above example, the Class B mask of 255.255.0.0 is implied. If you do not want to use the natural mask, you must enter a subnet mask. For example, to create a static route to IP address 10.255.11.0, you would have to enter the Class C mask of 255.255.255.0:

```
-> ip static-route 10.255.11.0 mask 255.255.255.0 gateway 171.11.2.1
```

Note that specifying the length of the mask in bits is also supported. For example, the above static route is also configurable using the following command:

```
-> ip static-route 10.255.11.0/24 gateway 171.11.2.1
```

When you create a static route, the default metric value of 1 is used. However, you can change the priority of the route by increasing its metric value. The lower the metric value, the higher the priority. This metric is added to the metric cost of the route. The metric range is 1 to 15. For example:

```
-> ip static-route 10.255.11.0/24 gateway 171.11.2.1 metric 5
```

Static routes do not age out of the IP Forwarding table; you must delete them from the table. Use the **no ip static route** command to delete a static route. You must specify the destination IP address of the route as well as the IP address of the first hop (gateway). For example, to delete a static route to IP address 171.11.0.0 through gateway 171.11.2.1, you would enter:

```
-> no ip static-route 171.11.0.0 gateway 171.11.2.1
```

The IP Forwarding table includes routes learned through RIP as well as any static routes that are configured. Use the **show ip route** command to display the IP Forwarding table.

Note. A static route is not active unless the gateway it is using is active.

Creating a Default Route

A default route can be configured for packets destined for networks that are unknown to the switch. Use the **ip static-route** command to create a default route. You must specify a default route of 0.0.0.0 with a subnet mask of 0.0.0.0 and the IP address of the next hop (gateway). For example, to create a default route through gateway 171.11.2.1 you would enter:

```
-> ip static-route 0.0.0.0 mask 0.0.0.0 gateway 171.11.2.1
```

Note that specifying the length of the mask in bits is also supported. For example, the above default route is also configurable using the following command:

```
-> ip static-route 0.0.0.0/0 gateway 171.11.2.1
```

Note. You cannot create a default route by using the EMP port as a gateway.

Configuring Address Resolution Protocol (ARP)

To send packets on a locally connected network, the switch uses ARP to match the IP address of a device with its physical (MAC) address. To send a data packet to a device with which it has not previously communicated, the switch first broadcasts an ARP request packet. The ARP request packet requests the Ethernet hardware address corresponding to an Internet address. All hosts on the receiving Ethernet receive the ARP request, but only the host with the specified IP address responds. If present and functioning, the host with the specified IP address responds with an ARP reply packet containing its hardware address. The switch receives the ARP reply packet, stores the hardware address in its ARP cache for future use, and begins exchanging packets with the receiving device.

The switch stores the hardware address in its ARP cache (ARP table). The table contains a listing of IP addresses and their corresponding translations to MAC addresses. Entries in the table are used to translate 32-bit IP addresses into 48-bit Ethernet or IEEE 802.3 hardware addresses. Dynamic addresses remain in the table until they time out. You can set this time-out value and you can also manually add or delete permanent addresses to/from the table.

Adding a Permanent Entry to the ARP Table

As described above, dynamic entries remain in the ARP table for a specified time period before they are automatically removed. However, you can create a permanent entry in the table.

Use the **arp** command to add a permanent entry to the ARP table. You must enter the IP address of the entry followed by its physical (MAC) address. For example, to create an entry for IP address 171.11.1.1 with a corresponding physical address of 00:05:02:c0:7f:11, you would enter:

```
-> arp 171.11.1.1 00:05:02:c0:7f:11
```

When you add an entry to the ARP table, the IP address and hardware address (MAC address) are *required*. Optionally, you may also specify:

- **Alias.** Use the **alias** keyword to specify that the switch will act as an alias (proxy) for this IP address. When the alias option is used, the switch responds to all ARP requests for the specified IP address with its own MAC address. Note that this option is not related to Proxy ARP as defined in RFC 925.

For example:

```
-> arp 171.11.1.1 00:05:02:c0:7f:11 alias
```

Use the **show arp** command to display the ARP table.

Note. Because most hosts support the use of address resolution protocols to determine and cache address information (called dynamic address resolution), you generally do not need to specify permanent ARP entries.

Deleting a Permanent Entry from the ARP Table

Permanent entries do not age out of the ARP table. Use the **no arp** command to delete a permanent entry from the ARP table. When deleting an ARP entry, you only need to enter the IP address. For example, to delete an entry for IP address 171.11.1.1, you would enter:

```
-> no arp 171.11.1.1
```

Use the **show arp** command to display the ARP table and verify that the entry was deleted.

Note. You can also use the **no arp** command to delete a dynamic entry from the table.

Clearing a Dynamic Entry from the ARP Table

Dynamic entries can be cleared using the **clear arp-cache** command. This command clears all dynamic entries. Permanent entries must be cleared using the **no arp** command.

Use the **show arp** command to display the table and verify that the table was cleared.

Note. Dynamic entries remain in the ARP table until they time out. If the switch does not receive data from a host for this user-specified time, the entry is removed from the table. If another packet is received from this host, the switch goes through the discovery process again to add the entry to the table. The switch uses the MAC Address table time-out value as the ARP time-out value. Use the **mac-address-table aging-time** command to set the time-out value.

Local Proxy ARP

The Local Proxy ARP feature is an extension of the Proxy ARP feature, but is enabled on an IP interface and applies to the VLAN bound to that interface. When Local Proxy ARP is enabled, all ARP requests received on VLAN member ports are answered with the MAC address of the IP interface that has Local Proxy ARP enabled. In essence, all VLAN traffic is now routed within the VLAN instead of bridged.

This feature is intended for use with port mapping applications where VLANs are one-port associations. This allows hosts on the port mapping device to communicate via the router. ARP packets are still bridged across multiple ports.

Note that Local Proxy ARP takes precedence over any switch-wide Proxy ARP or ARP function. In addition, it is not necessary to configure Proxy ARP in order to use Local Proxy ARP. The two features are independent of each other.

By default, Local Proxy ARP is disabled when an IP interface is created. To enable this feature, use the **ip interface** command. For example:

```
-> ip interface Accounting local-proxy-arp
```

Note that when Local Proxy ARP is enabled for any one IP router interface associated with a VLAN, the feature is applied to the entire VLAN. It is not necessary to enable it for each interface. However, if the IP interface that has this feature enabled is moved to another VLAN, Local Proxy ARP is enabled for the new VLAN and must be enabled on another interface for the old VLAN.

ARP Filtering

ARP filtering is used to determine whether or not the switch responds to ARP requests that contain a specific IP address. This feature is generally used in conjunction with the Local Proxy ARP application; however, ARP filtering is available for use on its own and/or with other applications.

By default, no ARP filters exist in the switch configuration. When there are no filters present, all ARP packets are processed, unless they are blocked or redirected by some other feature.

Use the **ip dos arp-poison restricted-address** command to specify the following parameter values required to create an ARP filter:

- An IP address (e.g., 193.204.173.21) used to determine whether or not an ARP packet is filtered.
- An IP mask (e.g. 255.0.0.0) used to identify which part of the ARP packet IP address is compared to the filter IP address.
- An optional VLAN ID to specify that the filter is only applied to ARP packets from that VLAN.
- Which ARP packet IP address to use for filtering (sender or target). If the target IP address in the ARP packet matches a target IP specified in a filter, then the disposition for that filter applies to the ARP packet. If the sender IP address in the ARP packet matches a sender IP specified in a filter, then the disposition for that filter applies to the ARP packet.
- The filter disposition (block or allow). If an ARP packet meets filter criteria, the switch is either blocked from responding to the packet or allowed to respond to the packet depending on the filter disposition. Packets that do not meet any filter criteria are responded to by the switch.

The following **arp filter** command example creates an ARP filter, which will block the switch from responding to ARP packets that contain a sender IP address that starts with 198:

```
-> arp filter 198.0.0.0 mask 255.0.0.0 sender block
```

Up to 200 ARP filters can be defined on a single switch. To remove an individual filter, use the no form of the **arp filter** command. For example:

```
-> no arp filter 198.0.0.0
```

To clear all ARP filters from the switch configuration, use the **clear arp filter** command. For example:

```
-> clear arp filter
```

Use the **show arp filter** command to verify the ARP filter configuration. For more information about this and other ARP filter commands, see the *OmniSwitch 6250 CLI Reference Guide*.

IP Configuration

IP is enabled on the switch by default and there are few options that can, or need to be, configured. This section provides instructions for some basic IP configuration options.

Configuring the DHCP Client Interface

The `ip interface dhcp-client` command can be used to create a DHCP client interface on the switch. For example, to configure a DHCP client interface on VLAN 100 you would enter :

```
-> ip interface dhcp-client vlan 100
```

Refer to the [“Configuring the DHCP Client Interface” on page 29-12](#) for more detailed information regarding DHCP client.

Configuring the Router Primary Address

By default, the router primary address is derived from the first IP interface that becomes operational on the router. Use the `ip router primary-address` command to configure the router primary address. Enter the command, followed by the IP address. For example, to configure a router primary address of 172.22.2.115, you would enter:

```
-> ip router primary-address 172.22.2.115
```

Configuring the Router ID

By default, the router primary address of the router is used as the router ID. However, if a primary address has not been explicitly configured, the router ID defaults to the address of the first IP interface that becomes operational.

Use the `ip router router-id` command to configure the router ID. Enter the command, followed by the IP address. For example, to configure a router ID of 172.22.2.115, you would enter:

```
-> ip router router-id 172.22.2.115
```

Configuring the Route Preference of a Router

By default, the route preference of a router is in this order: local, static, and RIP (highest to lowest).

Use the `ip route-pref` command to change the route preference value of a router. For example, to configure the route preference of a RIP route, you would enter:

```
-> ip route-pref rip 15
```

To display the current route preference configuration, use the `show ip route-pref` command:

```
-> show ip route-pref
  Protocol      Route Preference Value
-----+-----
  Local         1
  Static        2
  RIP           120
```

Configuring the Time-to-Live (TTL) Value

The TTL value is the default value inserted into the TTL field of the IP header of datagrams originating from the switch whenever a TTL value is not supplied by the transport layer protocol. The value is measured in hops.

Use the **ip default-ttl** command to set the TTL value. Enter the command, followed by the TTL value. For example, to set a TTL value of 75, you would enter:

```
-> ip default-ttl 75
```

The default hop count is 64. The valid range is 1 to 255. Use the **show ip config** command to display the default TTL value.

Configuring Route Map Redistribution

It is possible to learn and advertise IPv4 routes between different protocols. Such a process is referred to as route redistribution and is configured using the **ip redistrib** command.

Redistribution uses route maps to control how external routes are learned and distributed. A route map consists of one or more user-defined statements that can determine which routes are allowed or denied access to the receiving network. In addition a route map may also contain statements that modify route parameters before they are redistributed.

When a route map is created, it is given a name to identify the group of statements that it represents. This name is required by the **ip redistrib** command. Therefore, configuring route redistribution involves the following steps:

- 1 Create a route map, as described in [“Using Route Maps” on page 25-16](#).
- 2 Configure redistribution to apply a route map, as described in [“Configuring Route Map Redistribution” on page 25-21](#).

Using Route Maps

A route map specifies the criteria that are used to control redistribution of routes between protocols. Such criteria is defined by configuring route map statements. There are three different types of statements:

- **Action.** An action statement configures the route map name, sequence number, and whether or not redistribution is permitted or denied based on route map criteria.
- **Match.** A match statement specifies criteria that a route must match. When a match occurs, then the action statement is applied to the route.
- **Set.** A set statement is used to modify route information before the route is redistributed into the receiving protocol. This statement is only applied if all the criteria of the route map is met and the action permits redistribution.

The **ip route-map** command is used to configure route map statements and provides the following **action**, **match**, and **set** parameters:

ip route-map action ...	ip route-map match ...	ip route-map set ...
permit	ip-address	metric
deny	ip-nexthop	tag
	ipv6-address	ip-nexthop
	ipv6-nexthop	ipv6-nexthop
	tag	
	ipv4-interface	
	ipv6-interface	
	metric	

Refer to the “IP Commands” chapter in the *OmniSwitch CLI Reference Guide* for more information about the **ip route-map** command parameters and usage guidelines.

Once a route map is created, it is then applied using the **ip redistrib** command. See “[Configuring Route Map Redistribution](#)” on page 25-21 for more information.

Creating a Route Map

When a route map is created, it is given a name (up to 20 characters), a sequence number, and an action (permit or deny). Specifying a sequence number is optional. If a value is not configured, then the number 50 is used by default.

To create a route map, use the **ip route-map** command with the **action** parameter. For example,

```
-> ip route-map static-to-rip sequence-number 10 action permit
```

The above command creates the static-to-rip route map, assigns a **sequence number** of 10 to the route map, and specifies a **permit** action.

To optionally filter routes before redistribution, use the **ip route-map** command with a **match** parameter to configure match criteria for incoming routes. For example,

```
-> ip route-map static-to-rip sequence-number 10 match tag 8
```

The above command configures a match statement for the static-to-rip route map to filter routes based on their tag value. When this route map is applied, only Static routes with a tag value of eight are redistributed into the RIP network. All other routes with a different tag value are dropped.

Note. Configuring match statements is not required. However, if a route map does not contain any match statements and the route map is applied using the **ip redistrib** command, the router redistributes *all* routes into the network of the receiving protocol.

To modify route information before it is redistributed, use the **ip route-map** command with a **set** parameter. For example,

```
-> ip route-map static-to-rip sequence-number 10 set tag 5
```

The above command configures a set statement for the static-to-rip route map that changes the route tag value to five. Because this statement is part of the static-to-rip route map, it is only applied to routes that have an existing tag value equal to eight.

The following is a summary of the commands used in the above examples:

```
-> ip route-map static-to-rip sequence-number 10 action permit
-> ip route-map static-to-rip sequence-number 10 match tag 8
-> ip route-map static-to-rip sequence-number 10 set tag 5
```

To verify a route map configuration, use the **show ip route-map** command:

```
-> show ip route-map
Route Maps: configured: 1 max: 200
Route Map: static-to-rip Sequence Number: 10 Action permit
  match tag 8
  set tag 5
```

Deleting a Route Map

Use the **no** form of the **ip route-map** command to delete an entire route map, a route map sequence, or a specific statement within a sequence.

To delete an entire route map, enter **no ip route-map** followed by the route map name. For example, the following command deletes the entire route map named `redistipv4`:

```
-> no ip route-map redistipv4
```

To delete a specific sequence number within a route map, enter **no ip route-map** followed by the route map name, then **sequence-number** followed by the actual number. For example, the following command deletes sequence 10 from the `redistipv4` route map:

```
-> no ip route-map redistipv4 sequence-number 10
```

Note that in the above example, the `redistipv4` route map is not deleted. Only those statements associated with sequence 10 are removed from the route map.

To delete a specific statement within a route map, enter **no ip route-map** followed by the route map name, then **sequence-number** followed by the sequence number for the statement, then either **match** or **set** and the match or set parameter and value. For example, the following command deletes only the match tag 8 statement from route map `redistipv4` sequence 10:

```
-> no ip route-map redistipv4 sequence-number 10 match tag 8
```

Configuring Route Map Sequences

A route map may consist of one or more sequences of statements. The sequence number determines which statements belong to which sequence and the order in which sequences for the same route map are processed.

To add match and set statements to an existing route map sequence, specify the same route map name and sequence number for each statement. For example, the following series of commands creates route map `rm_1` and configures match and set statements for the `rm_1` sequence 10:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 8
-> ip route-map rm_1 sequence-number 10 set metric 1
```

To configure a new sequence of statements for an existing route map, specify the same route map name but use a different sequence number. For example, the following commands create a new sequence 20 for the `rm_1` route map:

```
-> ip route-map rm_1 sequence-number 20 action permit
-> ip route-map rm_1 sequence-number 20 match ipv4-interface to-finance
-> ip route-map rm_1 sequence-number 20 set metric 5
```

The resulting route map appears as follows:

```
-> show ip route-map rm_1
Route Map: rm_1 Sequence Number: 10 Action permit
  match tag 8
  set metric 1
Route Map: rm_1 Sequence Number: 20 Action permit
  match ip4 interface to-finance
  set metric 5
```

Sequence 10 and sequence 20 are both linked to route map `rm_1` and are processed in ascending order according to their sequence number value. Note that there is an implied logical OR between sequences. As a result, if there is no match for the tag value in sequence 10, then the match interface statement in sequence 20 is processed. However, if a route matches the tag 8 value, then sequence 20 is not used. The set statement for whichever sequence was matched is applied.

A route map sequence may contain multiple match statements. If these statements are of the same kind (e.g., match tag 5, match tag 8, etc.) then a logical OR is implied between each like statement. If the match statements specify different types of matches (e.g. match tag 5, match ip4 interface to-finance, etc.), then a logical AND is implied between each statement. For example, the following route map sequence will redistribute a route if its tag is either 8 or 5:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 5
-> ip route-map rm_1 sequence-number 10 match tag 8
```

The following route map sequence will redistribute a route if the route has a tag of 8 or 5 *and* the route was learned on the IPv4 interface to-finance:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 5
-> ip route-map rm_1 sequence-number 10 match tag 8
-> ip route-map rm_1 sequence-number 10 match ipv4-interface to-finance
```

Configuring Access Lists

An IP access list provides a convenient way to add multiple IPv4 or IPv6 addresses to a route map. Using an access list avoids having to enter a separate route map statement for each individual IP address. Instead, a single statement is used that specifies the access list name. The route map is then applied to all the addresses contained within the access list.

Configuring an IP access list involves two steps: creating the access list and adding IP addresses to the list. To create an IP access list, use the **ip access-list** command (IPv4) or the **ipv6 access-list** command (IPv6) and specify a name to associate with the list. For example,

```
-> ip access-list ipaddr
-> ipv6 access-list ip6addr
```

To add addresses to an access list, use the **ip access-list address** (IPv4) or the **ipv6 access-list address** (IPv6) command. For example, the following commands add addresses to an existing access list:

```
-> ip access-list ipaddr address 10.0.0.0/8
-> ipv6 access-list ip6addr address 2001::/64
```

Use the same access list name each time the above commands are used to add additional addresses to the same access list. In addition, both commands provide the ability to configure if an address and/or its matching subnet routes are permitted (the default) or denied redistribution. For example:

```
-> ip access-list ipaddr address 16.24.2.1/16 action deny redistrib-control all-
subnets
-> ipv6 access-list ip6addr address 2001::1/64 action permit redistrib-control no-
subnets
```

For more information about configuring access list commands, see the “IP Commands” chapter in the *OmniSwitch CLI Reference Guide*.

Configuring Route Map Redistribution

The **ip redistrib** command is used to configure the redistribution of routes from a source protocol into the destination protocol. This command is used on the IPv4 router that will perform the redistribution.

A source protocol is a protocol from which the routes are learned. A destination protocol is the one into which the routes are redistributed. Make sure that both protocols are loaded and enabled before configuring redistribution.

Redistribution applies criteria specified in a route map to routes received from the source protocol. Therefore, configuring redistribution requires an existing route map. For example, the following command configures the redistribution of Static routes into a RIP network using the static-to-rip route map:

```
-> ip redistrib static into rip route-map static-to-rip
```

Static routes received by the router interface are processed based on the contents of the static-to-rip route map. Routes that match criteria specified in this route map are either allowed or denied redistribution into the RIP network. The route map may also specify the modification of route information before the route is redistributed. See [“Using Route Maps” on page 25-16](#) for more information.

To remove a route map redistribution configuration, use the **no** form of the **ip redistrib** command. For example:

```
-> no ip redistrib static into rip route-map static-to-rip
```

Use the **show ip redistrib** command to verify the redistribution configuration:

```
-> show ip redistrib
```

Source Protocol	Destination Protocol	Status	Route Map
LOCAL4	RIP	Enabled	rip_1

Configuring the Administrative Status of the Route Map Redistribution

The administrative status of a route map redistribution configuration is enabled by default. To change the administrative status, use the **status** parameter with the **ip redistrib** command. For example, the following command disables the redistribution administrative status for the specified route map:

```
-> ip redistrib static into rip route-map static-to-rip status disable
```

The following command example enables the administrative status:

```
-> ip redistrib static into rip route-map static-to-rip status enable
```

Route Map Redistribution Example

The following example configures the redistribution of Static routes into a RIP network using a route map (static-to-rip) to filter specific routes:

```
-> ip route-map static-to-rip sequence-number 10 action deny
-> ip route-map static-to-rip sequence-number 10 match tag 5

-> ip route-map static-to-rip sequence-number 20 action permit
-> ip route-map static-to-rip sequence-number 20 match ipv4-interface
intf_static
-> ip route-map static-to-rip sequence-number 20 set metric 255

-> ip route-map static-to-rip sequence-number 30 action permit
-> ip route-map static-to-rip sequence-number 30 set tag 8

-> ip redist static into rip route-map static-to-rip
```

The resulting static-to-rip route map redistribution configuration does the following

- Denies the redistribution of routes with a tag set to five.
- Redistributes into RIP all routes learned on the intf_rip interface and sets the metric for such routes to 255.
- Redistributes into RIP all other routes (those not processed by sequence 10 or 20) and sets the tag for such routes to eight.

IP-Directed Broadcasts

An IP directed broadcast is an IP datagram that has all zeroes or all 1 in the host portion of the destination IP address. The packet is sent to the broadcast address of a subnet to which the sender is not directly attached. Directed broadcasts are used in denial-of-service “smurf” attacks. In a smurf attack, a continuous stream of ping requests is sent from a falsified source address to a directed broadcast address, resulting in a large stream of replies, which can overload the host of the source address. By default, the switch drops directed broadcasts. Typically, directed broadcasts should not be enabled.

Use the `ip directed-broadcast` command to enable or disable IP-directed broadcasts. For example:

```
-> ip directed-broadcast off
```

Use the `show ip config` command to display the IP-directed broadcast state.

Denial of Service (DoS) Filtering

By default, the switch filters denial of service (DoS) attacks, which are security attacks aimed at devices that are available on a private network or the Internet. Some of these attacks aim at system bugs or vulnerability (for example, teardrop attacks), while other types of attacks involve generating large volumes of traffic so that network service will be denied to legitimate network users (such as peps attacks). These attacks include the following:

- **ICMP Ping of Death**—Ping packets that exceed the largest IP datagram size (65535 bytes) are sent to a host and hang or crash the system.
- **SYN Attack**—Floods a system with a series of TCP SYN packets, resulting in the host issuing SYN-ACK responses. The half open TCP connections can exhaust TCP resources, such that no other TCP connections are accepted.
- **Land Attack**—Spoofed packets are sent with the SYN flag set to a host on any open port that is listening. The machine may hang or reboot in an attempt to respond.
- **Teardrop/Bonk/Boink Attacks**—Bonk/boink/teardrop attacks generate IP fragments in a special way to exploit IP stack vulnerabilities. If the fragments overlap the way those attacks generate packets, an attack is recorded. Since teardrop, bonk, and boink all use the same IP fragmentation mechanism to attack, there is no distinction between detection of these attacks. The old IP fragments in the fragmentation queue is also reaped once the reassemble queue goes above certain size.
- **Pepsi Attack**—The most common form of UDP flooding directed at harming networks. A pepsi attack is an attack consisting of a large number of spoofed UDP packets aimed at diagnostic ports on network devices. This can cause network devices to use up a large amount of CPU time responding to these packets.
- **ARP Flood Attack**—Floods a switch with a large number of ARP requests, resulting in the switch using a large amount of the CPU time to respond to these requests. If the number of ARP requests exceeds the preset value of 500 per second, an attack is detected.

- **Invalid IP Attack**—Packets with invalid source or destination IP addresses are received by the switch. When such an Invalid-IP attack is detected, the packets are dropped, and SNMP traps are generated. Examples of some invalid source and destination IP addresses are listed below:

Invalid Source IP address	<ul style="list-style-type: none"> • 0.x.x.x. • 255.255.255.255. • subnet broadcast, i.e. 172.28.255.255, for an existing IP interface 172.28.0.0/16. • in the range 224.x.x.x - 255.255.255.254. • Source IP address equals one of Switch IP Interface addresses.
Invalid Destination IP address	<ul style="list-style-type: none"> • 127.x.x.x. • in the range 240.x.x.x - 255.255.255.254. • 0.0.0.0 (valid exceptions - certain DHCP packets e.g.). • 172.28.0.0 for a router network 172.28.4.11/16. • 0.x.x.x.

- **Multicast IP and MAC Address Mismatch**—This attack is detected when:
 - the source MAC address of a packet received by a switch is a Multicast MAC address.
 - the destination IP and MAC addresses of a packet received by a switch is same as the Multicast IP and MAC addresses, but the Multicast IP and the Multicast MAC addresses do not match.

Note. In both the conditions described above in “Multicast IP and MAC Address Mismatch”, packets are dropped and SNMP traps are generated.

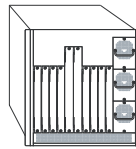
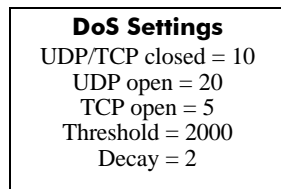
- the destination IP is a unicast IP and the destination MAC address is either a Broadcast or Multicast address. In such a condition, an event is recorded in the DoS statistics. No SNMP traps are generated because valid packets can also fall under this category.
- **Ping overload**—Floods a switch with a large number of ICMP packets, resulting in the switch using a large amount of CPU time to respond to these packets. If the number of ICMP packets exceed 100 per second, a DoS attack is detected. By default, the detection of attack is disabled.
- **Packets with loopback source IP address**—Packets with an invalid source address of 127.0.0.0/8 (loopback network) are received by the switch. When such packets are detected, they are dropped, and SNMP traps are generated.

The switch can be set to detect various types of port scans by monitoring for TCP or UDP packets sent to open or closed ports. Monitoring is done in the following manner:

- **Packet penalty values set.** TCP and UDP packets destined for open or closed ports are assigned a penalty value. Each time a packet of this type is received, its assigned penalty value is added to a running total. This total is cumulative and includes all TCP and UDP packets destined for open or closed ports.

- **Port scan penalty value threshold.** The switch is given a port scan penalty value threshold. This number is the maximum value the running penalty total can achieve before triggering an SNMP trap.
- **Decay value.** A decay value is set. The running penalty total is divided by the decay value every minute.
- **Trap generation.** If the total penalty value exceeds the set port scan penalty value threshold, a trap is generated to alert the administrator that a port scan may be in progress.

For example, imagine that a switch is set so that TCP and UDP packets destined for closed ports are given a penalty of 10, TCP packets destined for open ports are given a penalty of 5, and UDP packets destined for open ports are given a penalty of 20. The decay is set to 2, and the switch port scan penalty value threshold is set to 2000:

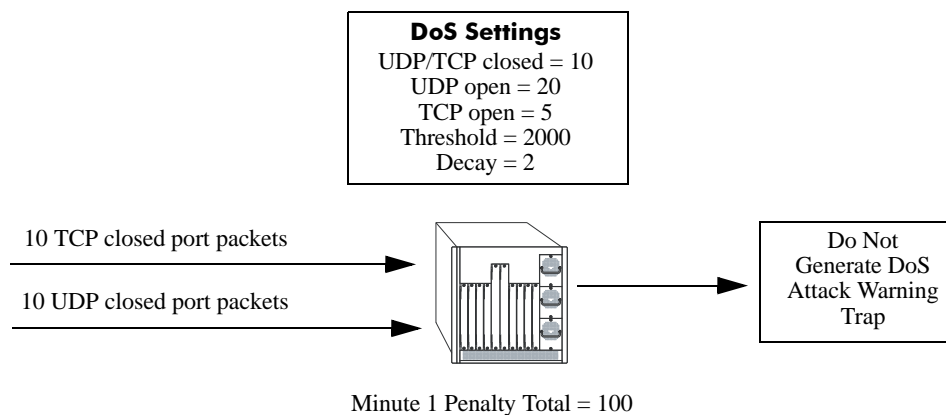


Penalty Total = 0

In one minute, 10 TCP closed port packets and 10 UDP closed port packets are received. This would bring the total penalty value to 200, as shown using the following equation:

$$(10 \text{ TCP} \times 10 \text{ penalty}) + (10 \text{ UDP} \times 10 \text{ penalty}) = 200$$

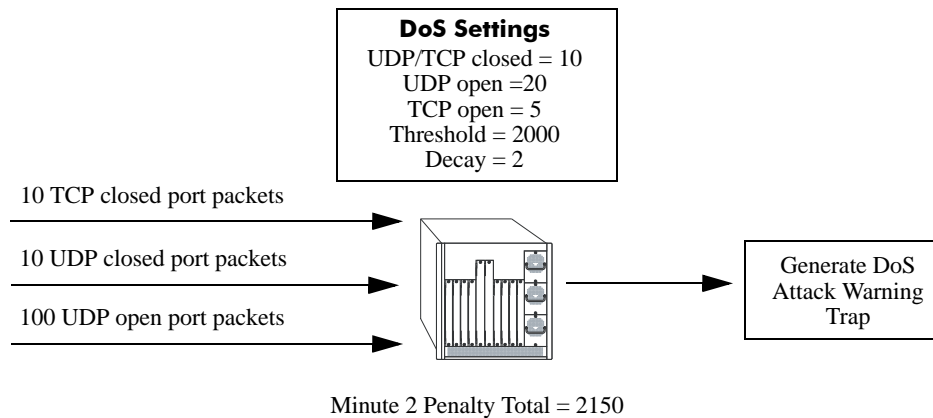
This value would be divided by 2 (due to the decay) and decreased to 100. The switch would not record a port scan:



In the next minute, 10 more TCP and UDP closed port packets are received, along with 200 UDP open-port packets. This would bring the total penalty value to 4300, as shown using the following equation:

$$(100 \text{ previous minute value}) + (10 \text{ TCP} \times 10 \text{ penalty}) + (10 \text{ UDP} \times 10 \text{ penalty}) + (200 \text{ UDP} \times 20 \text{ penalty}) = 4300$$

This value would be divided by 2 (due to decay) and decreased to 2150. The switch would record a port scan and generate a trap to warn the administrator:



The above functions and how to set their values are covered in the sections that follow.

Setting Penalty Values

There are three types of traffic you can set a penalty value for:

- TCP/UDP packets bound for closed ports.
- TCP traffic bound for open ports.
- UDP traffic bound for open ports.

Each type has its own command to assign a penalty value. Penalty values can be any non-negative integer. Each time a packet is received that matches an assigned penalty, the total penalty value for the switch is increased by the penalty value of the packet in question.

To assign a penalty value to TCP/UDP packets bound for a closed port, use the **ip dos scan close-port-penalty** command with a penalty value. For example, to assign a penalty value of 10 to TCP/UDP packets destined for closed ports, enter the following:

```
-> ip dos scan close-port-penalty 10
```

To assign a penalty value to TCP packets bound for an open port, use the **ip dos scan tcp open-port-penalty** command with a penalty value. For example, to assign a penalty value of 10 to TCP packets destined for opened ports, enter the following:

```
-> ip dos scan tcp open-port-penalty 10
```

To assign a penalty value to UDP packets bound for an open port, use the **ip dos scan udp open-port-penalty** command with a penalty value. For example, to assign a penalty value of 10 to TCP/UDP packets destined for closed ports, enter the following:

```
-> ip dos scan udp open-port-penalty 10
```

Setting the Port Scan Penalty Value Threshold

The port scan penalty value threshold is the highest point the total penalty value for the switch can reach before a trap is generated informing the administrator that a port scan is in progress.

To set the port scan penalty value threshold, enter the threshold value with the **ip dos scan threshold** command. For example, to set the port scan penalty value threshold to 2000, enter the following:

```
-> ip dos scan threshold 2000
```

Setting the Decay Value

The decay value is the amount the total penalty value is divided by every minute. As the switch records incoming UDP and TCP packets, it adds their assigned penalty values together to create the total penalty value for the switch. To prevent the switch from registering a port scan from normal traffic, the decay value is set to lower the total penalty value every minute to compensate from normal traffic flow.

To set the decay value, enter the decay value with the **ip dos scan decay** command. For example, to set the decay value to 2, enter the following:

```
-> ip dos scan decay 2
```

Enabling DoS Traps

DoS traps must be enabled in order for the switch to warn the administrator that a port scan may be in progress when the switch's total penalty value crosses the port scan penalty value threshold.

To enable SNMP trap generation, enter the **ip dos trap** command, as shown:

```
-> ip dos trap enable
```

To disable DoS traps, enter the same **ip dos trap** command, as shown:

```
-> ip dos trap disable
```

ARP Poisoning

ARP Poisoning allows an attacker to sniff and tamper the data frames on a network. It also modifies or halts the traffic. The principle of ARP Poisoning is to send false or spoofed ARP messages to an Ethernet LAN.

Alcatel-Lucent introduces the functionality that detects the presence of an ARP poisoning host on a network. This functionality uses a configured restricted IP addresses, so that the switch will not get ARP response on sending an ARP request. If an ARP response is received, then an event is logged and the user is alerted using an SNMP trap.

Use the **ip dos arp-poison restricted-address** command to add an ARP Poison restricted address. Enter the command, followed by the IP address. For example, to add an ARP Poison restricted address as 192.168.1.1, you would enter:

```
-> ip dos arp-poison restricted-address 192.168.1.1
```

A maximum of two IP addresses per IP interface can be configured as restricted addresses.

To delete an ARP Poison restricted address, enter **no ip dos arp-poison restricted-address** followed by the IP address. For example:

```
-> no ip dos arp-poison restricted-address 192.168.1.1
```

To verify the number of attacks detected for configured ARP poison restricted addresses, use the **show ip dos arp-poison** command. For more information about this command, see the *OmniSwitch CLI Reference Guide*.

Enabling/Disabling IP Services

When a switch initially boots up, all supported TCP/UDP well-known service ports are enabled (open). Although these ports provide access for essential switch management services, such as telnet, ftp, snmp, etc., they also are vulnerable to DoS attacks. It is possible to scan open service ports and launch such attacks based on well-known port information.

The **ip service** command allows you to selectively disable (close) TCP/UDP well-known service ports and enable them when necessary. This command only operates on TCP/UDP ports that are opened by default. It has no effect on ports that are opened by loading applications, such as RIP.

In addition, the **ip service** command allows you to designate which port to enable or disable by specifying the name of a service or the well-known port number associated with that service. For example, both of the following commands disable the telnet service:

```
-> no ip service telnet
-> no ip service port 23
```

Note that specifying a port number requires the use of the optional **port** keyword.

To enable or disable more than one service in a single command line, enter each service name separated by a space. For example, the following command enables the telnet, ftp, and snmp service ports:

```
-> ip service telnet ftp snmp
```

The following table lists **ip service** command options for specifying TCP/UDP services and also includes the well-known port number associated with each service:

service	port
ftp	21
ssh	22
telnet	23
http	80
secure-http	443
udp-relay	67
network-time	123
snmp	161
proprietary	1024
proprietary	1025

Managing IP

The following sections describe IP commands that can be used to monitor and troubleshoot IP forwarding on the switch.

Internet Control Message Protocol (ICMP)

Internet Control Message Protocol (ICMP) is a network layer protocol within the IP protocol suite that provides message packets to report errors and other IP packet processing information back to the source. ICMP generates several kinds of useful messages, including Destination Unreachable, Echo Request and Reply, Redirect, Time Exceeded, and Router Advertisement and Solicitation. If an ICMP message cannot be delivered, a second one is not generated. This prevents an endless flood of ICMP messages.

When an ICMP destination-unreachable message is sent by a switch, it means that the switch is unable to send the package to its final destination. The switch then discards the original packet. There are two reasons why a destination might be unreachable. Most commonly, the source host has specified a non-existent address. Less frequently, the switch does not have a route to the destination. The destination-unreachable messages include four basic types:

- **Network-Unreachable Message**—Usually means that a failure has occurred in the route lookup of the destination IP in the packet.
- **Host-Unreachable Message**—Usually indicates delivery failure, such as an unresolved client's hardware address or an incorrect subnet mask.
- **Protocol-Unreachable Message**—Usually means that the destination does not support the upper-layer protocol specified in the packet.
- **Port-Unreachable Message**—Implies that the TCP/UDP socket or port is not available.

Additional ICMP messages include:

- **Echo-Request Message**—Generated by the ping command, the message is sent by any host to test node reachability across an internetwork. The ICMP echo-reply message indicates that the node can be successfully reached.
- **Redirect Message**—Sent by the switch to the source host to stimulate more efficient routing. The switch still forwards the original packet to the destination. ICMP redirect messages allow host routing tables to remain small because it is necessary to know the address of only one switch, even if that switch does not provide the best path. Even after receiving an ICMP redirect message, some devices might continue using the less-efficient route.
- **Time-Exceeded Message**—Sent by the switch if an IP packet's TTL field reaches zero. The TTL field prevents packets from continuously circulating the internetwork if the internetwork contains a routing loop. Once a packet's TTL field reaches 0, the switch discards the packet.

Activating ICMP Control Messages

ICMP messages are identified by a *type* and a *code*. This number pair specifies an ICMP message. By default, ICMP messages are disabled. For example, ICMP type 4, code 0, specifies the source quench ICMP message.

To enable or disable an ICMP message, use the **icmp type** command with the type and code. For example, to enable the source quench the ICMP message (type 4, code 0) enter the following:

```
-> icmp type 4 code 0 enable
```

The following table is provide to identify the various ICMP messages, and their type and code:

ICMP Message	Type	Code
echo reply	0	0
network unreachable	0	3
host unreachable	3	1
protocol unreachable	3	2
port unreachable	3	3
frag needed but DF bit set	3	4
source route failed	3	5
destination network unknown	3	6
destination host unknown	3	7
source host isolated	3	8
dest network admin prohibited	3	9
host admin prohibited by filter	3	10
network unreachable for TOS	3	11
host unreachable for TOS	3	12
source quench	4	0
redirect for network	5	0
redirect for host	5	1
redirect for TOS and network	5	2
redirect for TOS and host	5	3
echo request	8	0
router advertisement	9	0
router solicitation	10	0
time exceeded during transmit	11	0
time exceeded during reassembly	11	1
ip header bad	12	0
required option missing	12	1
timestamp request	13	0
timestamp reply	14	0
information request (obsolete)	15	0
information reply (obsolete)	16	0
address mask request	17	0

ICMP Message	Type	Code
address mask reply	18	0

In addition to the **icmp type** command, several commonly used ICMP messages have been separate CLI commands for convenience. These commands are listed below with the ICMP message name, type, and code:

ICMP Message	Command
Network unreachable (type 0, code 3)	icmp unreachable
Host unreachable (type 3, code 1)	icmp unreachable
Protocol unreachable (type 3, code 2)	icmp unreachable
Port unreachable (type 3, code 3)	icmp unreachable
Echo reply (type 0, code 0)	icmp echo
Echo request (type 8, code 0)	icmp echo
Timestamp request (type 13, code 0)	icmp timestamp
Timestamp reply (type 14, code 0)	icmp timestamp
Address Mask request (type 17, code 0)	icmp addr-mask
Address Mask reply (type 18, code 0)	icmp addr-mask

These commands are entered as the **icmp type** command, only without specifying a type or code. The echo, timestamp, and address mask commands have options for distinguishing between a request or a reply, and the unreachable command has options distinguishing between a network, host, protocol, or port.

For example, to enable an echo request message, enter the following:

```
-> icmp echo request enable
```

To enable a network unreachable message, enter the following:

```
-> icmp unreachable net-unreachable enable
```

Note. Enabling **host-unreachable** and **net-unreachable** messages are not recommended as it can cause the switch instability due to high-CPU conditions depending upon the volume of traffic required by these messages.

See [Chapter 14, “IP Commands,”](#) for specifics on the ICMP message commands.

Enabling All ICMP Types

To enable all ICMP message types, use the **icmp messages** command with the **enable** keyword. For example:

```
-> icmp messages enable
```

To disable all ICMP messages, enter the same command with the **disable** keyword. For example:

```
-> icmp messages enable
```

Setting the Minimum Packet Gap

The minimum packet gap is the time required between sending messages of a like type. For instance, if the minimum packet gap for Address Mask request messages is 40 microseconds, and an Address Mask message is sent, at least 40 microseconds must pass before another one could be sent.

To set the minimum packet gap, use the **min-pkt-gap** keyword with any of the ICMP control commands. For example, to set the Source Quench minimum packet gap to 100 microseconds, enter the following:

```
-> icmp type 4 code 0 min-pkt-gap 100
```

Likewise, to set the Timestamp Reply minimum packet gap to 100 microseconds, enter the following:

```
-> icmp timestamp reply min-pkt-gap 100
```

The default minimum packet gap for ICMP messages is 0.

ICMP Control Table

The ICMP Control Table displays the ICMP control messages, whether they are enabled or disabled, and the minimum packet gap times. Use the **show icmp control** command to display the table.

ICMP Statistics Table

The ICMP Statistics Table displays the ICMP statistics and errors. This data can be used to monitor and troubleshoot IP on the switch. Use the **show icmp statistics** command to display the table.

Using the Ping Command

The **ping** command is used to test whether an IP destination can be reached from the local switch. This command sends an ICMP echo request to a destination and then waits for a reply. To ping a destination, enter the **ping** command and enter either the destination's IP address or host name. The switch will ping the destination by using the default frame count, packet size, interval, and time-out parameters (6 frames, 64 bytes, 1 second, and 5 seconds, respectively). For example:

```
-> ping 172.22.2.115
```

When you ping a device, the device IP address or host name is required. Optionally, you may also specify:

- **Count.** Use the **count** keyword to set the number of frames to be transmitted.
- **Size.** Use the **size** keyword to set the size, in bytes, of the data portion of the packet sent for this ping. You can specify a size or a range of sizes up to 60000.
- **Interval.** Use the **interval** keyword to set the frequency, in seconds, that the switch will poll the host.
- **Time-out.** Use the time-out keyword to set the number of seconds the program will wait for a response before timing out.

For example, to send a ping with a count of 2, a size of 32 bytes, an interval of 2 seconds, and a time-out of 10 seconds you would enter:

```
-> ping 172.22.2.115 count 2 size 32 interval 2 timeout 10
```

Note. If you change the default values, they will only apply to the current ping. The next time you use the **ping** command, the default values will be used unless you enter different values again.

Tracing an IP Route

The **tracert** command is used to find the path taken by an IP packet from the local switch to a specified destination. This command displays the individual hops to the destination as well as some timing information. When using this command, you must enter the name of the destination as part of the command line (either the IP address or host name). Use the optional **max-hop** parameter to set a maximum hop count to the destination. If the trace reaches this maximum hop count without reaching the destination, the trace stops.

For example, to perform a traceroute to a device with an IP address of 172.22.2.115 with a maximum hop count of 10 you would enter:

```
-> traceroute 172.22.2.115 max-hop 10
```

Displaying TCP Information

Use the **show tcp statistics** command to display TCP statistics. Use the **show tcp ports** command to display TCP port information.

Displaying UDP Information

UDP is a secondary transport-layer protocol that uses IP for delivery. UDP is not connection-oriented and does not provide reliable end-to-end delivery of datagrams. But some applications can safely use UDP to send datagrams that do not require the extra overhead added by TCP. Use the **show udp statistics** command to display UDP statistics. Use the **show udp ports** command to display UDP port information.

Verifying the IP Configuration

A summary of the show commands used for verifying the IP configuration is given here:

show ip interface	Displays the usability status of interfaces configured for IP.
show ip route	Displays the IP Forwarding table.
show ip route-pref	Displays the configured route preference of a router.
show ip router database	Displays a list of all routes (static and dynamic) that exist in the IP router database.
show ip config	Displays IP configuration parameters.
show ip protocols	Displays switch routing protocol information and status.
show ip service	Displays the current status of TCP/UDP service ports. Includes service name and well-known port number.
show arp	Displays the ARP table.
show arp filter	Displays the ARP filter configuration for the switch.
show icmp control	This command allows the viewing of the ICMP control settings.
show ip dos config	Displays the configuration parameters of the DoS scan for the switch.
show ip dos statistics	Displays the statistics on detected port scans for the switch.
show ip dos arp-poison	Displays the number of attacks detected for a restricted address.

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

26 Configuring IPv6

Internet Protocol version 6 (IPv6) is the next generation of Internet Protocol version 4 (IPv4). Both versions are supported. Implementing IPv6 solves the limited address problem currently facing IPv4, which provides a 32-bit address space. IPv6 increases the address space available to 128 bits.

In This Chapter

This chapter describes IPv6 and how to configure it through Command Line Interface (CLI). The CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

This chapter provides an overview of IPv6 and includes information about the following procedures:

- Configuring an IPv6 interface (see [page 26-9](#))
- Assigning IPv6 Addresses (see [page 26-11](#))
- Creating a Static Route (see [page 26-13](#))
- Configuring the Route Preference of a Router (see [page 26-14](#))
- Configuring Route Map Redistribution (see [page 26-15](#))

IPv6 Specifications

Note that the maximum limit values provided in the following Specifications table are subject to available system resources:

RFCs Supported	2460– <i>Internet Protocol, Version 6 (IPv6) Specification</i> 2461– <i>Neighbor Discovery for IP Version 6 (IPv6)</i> 2462– <i>IPv6 Stateless Address Autoconfiguration</i> 2464– <i>Transmission of IPv6 Packets Over Ethernet Networks</i> 3056– <i>Connection of IPv6 Domains via IPv4 Clouds</i> 4213– <i>Basic Transition Mechanisms for IPv6 Hosts and Routers</i> 4291– <i>Internet Protocol Version 6 (IPv6) Addressing Architecture</i> 4443– <i>Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification</i>
Platforms Supported	OmniSwitch 6250 Series
Maximum IPv6 interfaces	16
Maximum IPv6 interfaces per VLAN	1
Maximum IPv6 global unicast addressess	16
Maximum IPv6 global unicast addresses per IPv6 interface	10
Maximum IPv6 static routes per switch	128
Maximum IPv6 host routes per switch	128
Maximum IPv6 neighbors (ND)	128
Maximum Number of RIPng Peers	10
Maximum Number of RIPng Interfaces	10
Maximum Number of RIPng Routes	128

IPv6 Defaults

The following table lists the defaults for IPv6 configuration through the **ip** command.

Description	Command	Default
Global status of IPv6 on the switch	N/A	Enabled
IPv6 interfaces	ipv6 interface	None

IPv6 Overview

IPv6 provides the basic functionality that is offered with IPv4 but includes the following enhancements and features not available with IPv4:

- **Increased IP address size**—IPv6 uses a 128-bit address, a substantial increase over the 32-bit IPv4 address size. Providing a larger address size also significantly increases the address space available, thus eliminating the concern over running out of IP addresses. See [“IPv6 Addressing” on page 26-6](#) for more information.
- **Autoconfiguration of addresses**—When an IPv6 interface is created or a device is connected to the switch, an IPv6 link-local address is automatically assigned for the interface and/or device. See [“Auto-configuration of IPv6 Addresses” on page 26-8](#) for more information.
- **Anycast addresses**—A new type of address. Packets sent to an anycast address are delivered to one member of the anycast group.
- **Simplified header format**—A simpler IPv6 header format is used to keep the processing and bandwidth cost of IPv6 packets as low as possible. As a result, the IPv6 header is only twice the size of the IPv4 header despite the significant increase in address size.
- **Improved support for header options**—Improved header option encoding allows more efficient forwarding, fewer restrictions on the length of options, and greater flexibility to introduce new options.
- **Security improvements**—Extension definitions provide support for authentication, data integrity, and confidentiality.
- **Neighbor Discovery protocol**—A protocol defined for IPv6 that detects neighboring devices on the same link and the availability of those devices. Additional information that is useful for facilitating the interaction between devices on the same link is also detected (e.g., neighboring address prefixes, address resolution, duplicate address detection, link MTU, and hop limit values, etc.).

This implementation of IPv6 also provides the following mechanisms to maintain compatibility between IPv4 and IPv6:

- Dual-stack support for both IPv4 and IPv6 on the same switch.
- Configuration of IPv6 and IPv4 interfaces on the same VLAN.
- Embedded IPv4 addresses in the four lower-order bits of the IPv6 address.

The remainder of this section provides a brief overview of the new IPv6 address notation and autoconfiguration of addresses.

IPv6 Addressing

One of the main differences between IPv6 and IPv4 is that the address size has increased from 32 bits to 128 bits. Going to a 128-bit address also increases the size of the address space to the point where running out of IPv6 addresses is not a concern.

The following types of IPv6 addresses are supported:

Link-local—A link-local address is a private unicast address that identifies an interface or device on the local network. This type of address allows communication with devices and/or neighboring nodes that are attached to the same physical link. Note that when the communication is between two nodes that are not attached to the same link, both nodes must have a configured global unicast address. Routing between link-local addresses is not available because link-local addresses are not known or advertised to the general network.

Unicast—Standard unicast addresses, similar to IPv4.

Multicast—Addresses that represent a group of devices. Traffic sent to a multicast address is delivered to all members of the multicast group.

Anycast—Traffic that is sent to this type of address is delivered to one member of the anycast group. The device that receives the traffic is usually the one that is easiest to reach as determined by the active routing protocol.

Note. IPv6 does not support the use of broadcast addresses. This functionality is replaced using improved multicast addressing capabilities.

IPv6 address types are identified by the high-order bits of the address, as shown in the following table:

Address Type	Binary Prefix	IPv6 Notation
Unspecified	00...0 (128 bits)	::/128
Loopback	00...1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-local unicast	111111010	FE80::/10
Global unicast	everything else	

Note that anycast addresses are unicast addresses that are not identifiable by a known prefix.

IPv6 Address Notation

IPv4 addresses are expressed using dotted decimal notation and consist of four eight-bit octets. If this same method was used for IPv6 addresses, the address would contain 16 such octets, thus making it difficult to manage. IPv6 addresses are expressed using *colon hexadecimal notation* and consist of eight 16-bit words, as shown in the following example:

```
1234:000F:531F:4567:0000:0000:BCD2:F34A
```

Note that any field may contain all zeros or all ones. In addition, it is possible to shorten IPv6 addresses by suppressing leading zeros. For example:

```
1234:F:531F:4567:0:0:BCD2:F34A
```

Another method for shortening IPv6 addresses is known as *zero compression*. When an address contains contiguous words that consist of all zeros, a double colon (::) is used to identify these words. For example, using zero compression the address 0:0:0:0:1234:531F:BCD2:F34A is expressed as follows:

```
::1234:531F:BCD2:F34A
```

Because the last four words of the above address are uncompressed values, the double colon indicates that the first four words of the address all contain zeros. Note that using the double colon is only allowed once within a single address. So if the address was 1234:531F:0:0:BCD2:F34A:0:0, a double colon could *not* replace both sets of zeros. For example, the first two versions of this address shown below are valid, but the last version is not valid:

- 1 1234:531F::BCD2:F34A:0:0
- 2 1234:531F:0:0:BCD2:F34A::
- 3 1234:531F::BCD2:F34A:: (not valid)

With IPv6 addresses that have long strings of zeros, the benefit of zero compression is more dramatic. For example, address FF00:0:0:0:0:4501:32 becomes FF00::4501:32.

Note that hexadecimal notation used for IPv6 addresses resembles the notation which is used for MAC addresses. However, it is important to remember that IPv6 addresses still identify a device at the Layer 3 level and MAC addresses identify a device at the Layer 2 level.

Another supported IPv6 address notation includes embedding an IPv4 address as the four lower-order bits of the IPv6 address. This is especially useful when dealing with a mixed IPv4/IPv6 network. For example:

```
0:0:0:0:0:212.100.13.6
```

IPv6 Address Prefix Notation

The Classless Inter-Domain Routing (CIDR) notation is used to express IPv6 address prefixes. This notation consists of the 128-bit IPv6 address followed by a slash (/) and a number representing the prefix length (IPv6-address/prefix-length). For example, the following IPv6 address has a prefix length of 64 bits:

```
FE80::2D0:95FF:FE12:FAB2/64
```

Autoconfiguration of IPv6 Addresses

This implementation of IPv6 supports the *stateless* autoconfiguration of link-local addresses for IPv6 VLAN interfaces and for devices when they are connected to the switch. Stateless refers to the fact that little or no configuration is required to generate such addresses and there is no dependency on an address configuration server, such as a DHCP server, to provide the addresses.

A link-local address is a private unicast address that identifies an interface or device on the local network. This type of address allows communication with devices and/or neighboring nodes that are attached to the same physical link. Note that when the communication is between two nodes that are not attached to the same link, both nodes must have a configured global unicast address. Routing between link-local addresses is not available because link-local addresses are not known or advertised to the general network.

When an IPv6 VLAN interface is created or a device is connected to the switch, a link-local address is automatically generated for the interface or device. This type of address consists of the well-known IPv6 prefix FE80::/64 combined with an interface ID. The interface ID is derived from the router MAC address associated with the IPv6 interface or the source MAC address if the address is for a device. The resulting link-local address resembles the following example:

```
FE80::2d0:95ff:fe6b:5ccd/64
```

Note that when this example address was created, the MAC address was modified by complementing the second bit of the leftmost byte and by inserting the hex values 0xFF and 0xFE between the third and fourth octets of the address. These modifications were made because IPv6 requires an interface ID that is derived using Modified EUI-64 format.

Stateless autoconfiguration is not available for assigning a global unicast or anycast address to an IPv6 interface. In other words, manual configuration is required to assign a non-link-local address to an interface. See [“Assigning IPv6 Addresses” on page 22-14](#) for more information.

Both stateless and *stateful* autoconfiguration is supported for devices, such as a workstation, when they are connected to the switch. When the stateless method is used in this instance, the device listens for router advertisements in order to obtain a subnet prefix. The unicast address for the device is then formed by combining the subnet prefix with the interface ID for that device.

Stateful autoconfiguration refers to the use of an independent server, such as a DHCP server, to obtain an IPv6 unicast address and other related information. Of course, manual configuration of an IPv6 address is always available for devices as well.

Regardless of how an IPv6 address is obtained, duplicate address detection (DAD) is performed before the address is assigned to an interface or device. If a duplicate is found, the address is not assigned. Note that DAD is *not* performed for anycast addresses.

Please refer to RFCs 2462, 2464, and 3513 for more technical information about autoconfiguration and IPv6 address notation.

Configuring an IPv6 Interface

The **ipv6 interface** command is used to create an IPv6 interface for a VLAN. Note the following when configuring an IPv6 interface:

- A unique interface name is required for a VLAN interface.
- If creating a VLAN interface, the VLAN must already exist. See [Chapter 4, “Configuring VLANs,”](#) for more information.
- The following configurable interface parameters are set to their default values unless otherwise specified when the **ipv6 interface** command is used:

IPv6 interface parameters

ra-send	ra-retrans-timer
ra-max-interval	ra-default-lifetime
ra-managed-config-flag	ra-send-mtu
ra-other-config-flag	base-reachable-time
ra-reachable-time	

Refer to the **ipv6 interface** command page in the *OmniSwitch 6250 CLI Reference Guide* for more details regarding these parameters.

- Each VLAN can have one IPv6 interface. Configuring both an IPv4 and IPv6 interface on the same VLAN is allowed. Note that the VLAN interfaces of both types are not active until at least one port associated with the VLAN goes active.
- A link-local address is automatically configured for an IPv6 interface when the interface is configured. For more information regarding how this address is formed, see [“Autoconfiguration of IPv6 Addresses” on page 26-8.](#)
- Assigning more than one IPv6 address to a single IPv6 interface is allowed.
- Assigning the same link-local address to multiple interfaces is allowed. Each global unicast prefix, however, can only exist on one interface. For example, if an interface for a VLAN 100 is configured with an address 4100:1000::1/64, an interface for VLAN 200 cannot have an address 4100:1000::2/64.
- Each IPv6 interface anycast address must also have a unique prefix. However, multiple devices may share the same anycast address prefix to identify themselves as members of the anycast group.

To create an IPv6 interface for a VLAN, enter **ipv6 interface** followed by an interface name, then followed by a VLAN ID. For example, the following command creates an IPv6 interface for VLAN 200:

```
-> ipv6 interface v6if-v200 vlan 200
```

Use the **show ipv6 interface** command to verify the interface configuration for the switch. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Modifying an IPv6 Interface

The **ipv6 interface** command is also used to modify existing IPv6 interface parameter values. It is not necessary to first remove the interface and then create it again with the new values. The changes specified will overwrite existing parameter values. For example, the following command changes the router advertisement (RA) reachable time and the RA retransmit timer values for interface *v6if-v200*:

```
-> ipv6 interface v6if-v200 ra-reachable-time 60000 ra-retrans-time 2000
```

When an existing interface name is specified with the **ipv6 interface** command, the command modifies specified parameters for that interface. If an unknown interface name is entered along with an existing VLAN parameter, a new interface is created with the name specified.

Removing an IPv6 Interface

To remove an IPv6 interface from the switch configuration, use the **no** form of the **ipv6 interface** command. Note that it is only necessary to specify the name of the interface, as shown in the following example:

```
-> no ipv6 interface v6if-v200
```


Assigning IPv6 Addresses

As was previously mentioned, when an IPv6 interface is created for a VLAN, an IPv6 link-local address is automatically created for that interface. This is also true when a device, such as a workstation, is connected to the switch.

Link-local addresses, although private and non-routable, enable interfaces and workstations to communicate with other interfaces and workstations that are connected to the same link. This simplifies getting devices up and running on the local network. If this level of communication is sufficient, assigning additional addresses is not required.

If it is necessary to identify an interface or device to the entire network, or as a member of a particular group, or enable an interface to perform routing functions, then configuring additional addresses (e.g., global unicast or anycast) is required.

Use the **ipv6 address** command to manually assign addresses to an existing interface or device. For example, the following command assigns a global unicast address to the VLAN interface *v6if-v200*:

```
-> ipv6 address 4100:1000::20/64 v6if-v200
```

In the above example, 4100:1000:: is specified as the subnet prefix and 20 is the interface identifier. Note that the IPv6 address is expressed using CIDR notation to specify the prefix length. In the above example, /64 indicates a subnet prefix length of 64 bits.

To use the MAC address of an interface or device as the interface ID, specify the **eui-64** option with this command. For example:

```
-> ipv6 address 4100:1000::/64 eui-64 v6if-v200
```

The above command example creates address 4100:1000::2d0:95ff:fe12:fab2/64 for interface *v6if-v200*.

Note the following when configuring IPv6 addresses:

- It is possible to assign more than one address to a single interface.
- Any field of an address may contain all zeros or all ones. The exception to this is the interface identifier portion of the address, which cannot be all zeros. If the **eui-64** option is specified with the **ipv6 address** command, this is not an issue.
- The EUI-64 interface identifier takes up the last 64 bits of the 128-bit IPv6 address. If the subnet prefix combined with the EUI-64 interface ID is longer than 128 bits, an error occurs and the address is not created.
- A subnet router anycast address is automatically created when a global unicast address is assigned to an interface. The anycast address is derived from the global address by adding an interface ID of all zeros to the prefix of the global address. For example, the global address 4100:1000::20/64 generates the anycast address 4100:1000::/64.
- Devices, such as a PC, are eligible for stateless autoconfiguration of unicast addresses in addition to the link-local address. If this type of configuration is in use on the network, manual configuration of addresses is not required.
- IPv6 VLAN interfaces are only eligible for stateless autoconfiguration of their link-local addresses. Manual configuration of addresses is required for all additional addresses.

See “[IPv6 Addressing](#)” on page 26-6 for an overview of IPv6 address notation. Refer to RFC 4291 for more technical address information.

Removing an IPv6 Address

To remove an IPv6 address from an interface, use the **no** form of the **ipv6 address** command as shown:

```
-> no ipv6 address 4100:1000::20 v6if-v200
```

Note that the subnet router anycast address is automatically deleted when the last unicast address of the same subnet is removed from the interface.

Creating an IPv6 Static Route

Static routes are user-defined and carry a higher priority than routes created by dynamic routing protocols. That is, if two routes have the same metric value, the static route has the higher priority. Static routes allow you to define, or customize, an explicit path to an IPv6 network segment, which is then added to the IPv6 Forwarding table. Static routes can be created between VLANs to enable devices on these VLANs to communicate.

Use the **ipv6 static-route** command to create a static route. You must specify the destination IPv6 address of the route as well as the IPv6 address of the first hop (gateway) used to reach the destination. For example, to create a static route to IPv6 address 212:95:5::/64 through gateway fe80::2d0:95ff:fe6a:f458 on interface v6if-137, you would enter:

```
-> ipv6 static-route 212:95:5::/64 gateway fe80::2d0:95ff:fe6a:f458 v6if-137
```

Note that in the example above the IPv6 interface name for the gateway was included. This parameter is required only when a link local address is specified as the gateway.

When you create a static route, the default metric value of 1 is used. However, you can change the priority of the route by increasing its metric value. The lower the metric value, the higher the priority. This metric is added to the metric cost of the route. The metric range is 1 to 15. For example:

```
-> ipv6 static-route 212:95:5::/64 gateway fe80::2d0:95ff:fe6a:f458 v6if-137 metric  
3
```

Static routes do not age out of the IPv6 Forwarding table; you must delete them from the table. Use the **no ipv6 static-route** command to delete a static route. You must specify the destination IPv6 address of the route as well as the IPv6 address of the first hop (gateway). For example, to delete a static route to IPv6 address 212:95:5::/64 through gateway fe80::2d0:95ff:fe6a:f458 on interface v6if-137, you would enter:

```
-> no ip static-route 212:95:5::/64 gateway fe80::2d0:95ff:fe6a:f458 v6if-137
```

The IPv6 Forwarding table includes routes learned through RIP as well as any static routes that are configured. Use the **show ipv6 routes** command to display the IPv6 Forwarding table.

Note. A static route is not active unless the gateway it is using is active.

Configuring the Route Preference of a Router

By default, the route preference of a router is in this order: local, static, and RIPng (highest to lowest).

Use the **ipv6 route-pref** command to change the route preference value of a router. For example, to configure the route preference of a RIP route, you would enter:

```
-> ipv6 route-pref rip 15
```

To display the current route preference configuration, use the **show ipv6 route-pref** command:

```
-> show ipv6 route-pref
  Protocol      Route Preference Value
-----+-----
  Local         1
  Static        2
  RIP           120
```

Configuring Route Map Redistribution

It is possible to learn and advertise IPv6 routes between different protocols. Such a process is referred to as route redistribution and is configured using the **ipv6 redistrib** command.

Redistribution uses route maps to control how external routes are learned and distributed. A route map consists of one or more user-defined statements that can determine which routes are allowed or denied access to the receiving network. In addition a route map may also contain statements that modify route parameters before they are redistributed.

When a route map is created, it is given a name to identify the group of statements that it represents. This name is required by the **ipv6 redistrib** command. Therefore, configuring route redistribution involves the following steps:

- 1 Create a route map, as described in [“Using Route Maps” on page 26-15](#).
- 2 Configure redistribution to apply a route map, as described in [“Configuring Route Map Redistribution” on page 26-19](#).

Using Route Maps

A route map specifies the criteria that are used to control redistribution of routes between protocols. Such criteria is defined by configuring route map statements. There are three different types of statements:

- **Action.** An action statement configures the route map name, sequence number, and whether or not redistribution is permitted or denied based on route map criteria.
- **Match.** A match statement specifies criteria that a route must match. When a match occurs, then the action statement is applied to the route.
- **Set.** A set statement is used to modify route information before the route is redistributed into the receiving protocol. This statement is only applied if all the criteria of the route map is met and the action permits redistribution.

The **ip route-map** command is used to configure route map statements and provides the following **action**, **match**, and **set** parameters:

ip route-map action ...	ip route-map match ...	ip route-map set ...
permit deny	ip-address ip-nexthop ipv6-address ipv6-nexthop tag ipv4-interface ipv6-interface metric	metric tag ip-nexthop ipv6-nexthop

Refer to the “IP Commands” chapter in the *OmniSwitch CLI Reference Guide* for more information about the **ip route-map** command parameters and usage guidelines.

Once a route map is created, it is then applied using the **ipv6 redistrib** command. See [“Configuring Route Map Redistribution” on page 26-19](#) for more information.

Creating a Route Map

When a route map is created, it is given a name (up to 20 characters), a sequence number, and an action (permit or deny). Specifying a sequence number is optional. If a value is not configured, then the number 50 is used by default.

To create a route map, use the **ip route-map** command with the **action** parameter. For example,

```
-> ip route-map static-to-rip sequence-number 10 action permit
```

The above command creates the static-to-rip route map, assigns a **sequence number** of 10 to the route map, and specifies a **permit** action.

To optionally filter routes before redistribution, use the **ip route-map** command with a **match** parameter to configure match criteria for incoming routes. For example,

```
-> ip route-map static-to-rip sequence-number 10 match tag 8
```

The above command configures a match statement for the static-to-rip route map to filter routes based on their tag value. When this route map is applied, only Static routes with a tag value of eight are redistributed into the RIP network. All other routes with a different tag value are dropped.

Note. Configuring match statements is not required. However, if a route map does not contain any match statements and the route map is applied using the **ipv6 redistrib** command, the router redistributes *all* routes into the network of the receiving protocol.

To modify route information before it is redistributed, use the **ip route-map** command with a **set** parameter. For example,

```
-> ip route-map static-to-rip sequence-number 10 set tag 5
```

The above command configures a set statement for the static-to-rip route map that changes the route tag value to five. Because this statement is part of the static-to-rip route map, it is only applied to routes that have an existing tag value equal to eight.

The following is a summary of the commands used in the above examples:

```
-> ip route-map static-to-rip sequence-number 10 action permit
-> ip route-map static-to-rip sequence-number 10 match tag 8
-> ip route-map static-to-rip sequence-number 10 set tag 5
```

To verify a route map configuration, use the **show ip route-map** command:

```
-> show ip route-map
Route Maps: configured: 1 max: 200
Route Map: static-to-rip Sequence Number: 10 Action permit
      match tag 8
      set tag 5
```

Deleting a Route Map

Use the **no** form of the **ip route-map** command to delete an entire route map, a route map sequence, or a specific statement within a sequence.

To delete an entire route map, enter **no ip route-map** followed by the route map name. For example, the following command deletes the entire route map named `redistipv4`:

```
-> no ip route-map redistipv4
```

To delete a specific sequence number within a route map, enter **no ip route-map** followed by the route map name, then **sequence-number** followed by the actual number. For example, the following command deletes sequence 10 from the `redistipv4` route map:

```
-> no ip route-map redistipv4 sequence-number 10
```

Note that in the above example, the `redistipv4` route map is not deleted. Only those statements associated with sequence 10 are removed from the route map.

To delete a specific statement within a route map, enter **no ip route-map** followed by the route map name, then **sequence-number** followed by the sequence number for the statement, then either **match** or **set** and the match or set parameter and value. For example, the following command deletes only the match tag 8 statement from route map `redistipv4` sequence 10:

```
-> no ip route-map redistipv4 sequence-number 10 match tag 8
```

Configuring Route Map Sequences

A route map may consist of one or more sequences of statements. The sequence number determines which statements belong to which sequence and the order in which sequences for the same route map are processed.

To add match and set statements to an existing route map sequence, specify the same route map name and sequence number for each statement. For example, the following series of commands creates route map `rm_1` and configures match and set statements for the `rm_1` sequence 10:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 8
-> ip route-map rm_1 sequence-number 10 set metric 1
```

To configure a new sequence of statements for an existing route map, specify the same route map name but use a different sequence number. For example, the following command creates a new sequence 20 for the `rm_1` route map:

```
-> ip route-map rm_1 sequence-number 20 action permit
-> ip route-map rm_1 sequence-number 20 match ipv4-interface to-finance
-> ip route-map rm_1 sequence-number 20 set metric 5
```

The resulting route map appears as follows:

```
-> show ip route-map rm_1
Route Map: rm_1 Sequence Number: 10 Action permit
  match tag 8
  set metric 1
Route Map: rm_1 Sequence Number: 20 Action permit
  match ip4 interface to-finance
  set metric 5
```

Sequence 10 and sequence 20 are both linked to route map `rm_1` and are processed in ascending order according to their sequence number value. Note that there is an implied logical OR between sequences. As a result, if there is no match for the tag value in sequence 10, then the match interface statement in sequence 20 is processed. However, if a route matches the tag 8 value, then sequence 20 is not used. The set statement for whichever sequence was matched is applied.

A route map sequence may contain multiple match statements. If these statements are of the same kind (e.g., match tag 5, match tag 8, etc.) then a logical OR is implied between each like statement. If the match statements specify different types of matches (e.g. match tag 5, match ip4 interface to-finance, etc.), then a logical AND is implied between each statement. For example, the following route map sequence will redistribute a route if its tag is either 8 or 5:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 5
-> ip route-map rm_1 sequence-number 10 match tag 8
```

The following route map sequence will redistribute a route if the route has a tag of 8 or 5 *and* the route was learned on the IPv6 interface to-finance:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 5
-> ip route-map rm_1 sequence-number 10 match tag 8
-> ip route-map rm_1 sequence-number 10 match ipv6-interface to-finance
```

Configuring Access Lists

An IP access list provides a convenient way to add multiple IPv4 or IPv6 addresses to a route map. Using an access list avoids having to enter a separate route map statement for each individual IP address. Instead, a single statement is used that specifies the access list name. The route map is then applied to all the addresses contained within the access list.

Configuring an IP access list involves two steps: creating the access list and adding IP addresses to the list. To create an IP access list, use the **ip access-list** command (IPv4) or the **ipv6 access-list** command (IPv6) and specify a name to associate with the list. For example,

```
-> ip access-list ipaddr
-> ipv6 access-list ip6addr
```

To add addresses to an access list, use the **ip access-list address** (IPv4) or the **ipv6 access-list address** (IPv6) command. For example, the following commands add addresses to an existing access list:

```
-> ip access-list ipaddr address 10.0.0.0/8
-> ipv6 access-list ip6addr address 2001::/64
```

Use the same access list name each time the above commands are used to add additional addresses to the same access list. In addition, both commands provide the ability to configure if an address and/or its matching subnet routes are permitted (the default) or denied redistribution. For example:

```
-> ip access-list ipaddr address 16.24.2.1/16 action deny redist-control all-
subnets
-> ipv6 access-list ip6addr address 2001::1/64 action permit redist-control no-
subnets
```

For more information about configuring access list commands, see the “IP Commands” chapter in the *OmniSwitch CLI Reference Guide*.

Configuring Route Map Redistribution

The **ipv6 redistrib** command is used to configure the redistribution of routes from a source protocol into the destination protocol. This command is used on the IPv6 router that will perform the redistribution.

Note. A router automatically becomes an Autonomous System Border Router (ASBR) when redistribution is configured on the router.

A source protocol is a protocol from which the routes are learned. A destination protocol is the one into which the routes are redistributed. Make sure that both protocols are loaded and enabled before configuring redistribution.

Redistribution applies criteria specified in a route map to routes received from the source protocol. Therefore, configuring redistribution requires an existing route map. For example, the following command configures the redistribution of Static routes into the RIPng network using the static-to-rip route map:

```
-> ipv6 redistrib static into rip route-map static-to-rip
```

Static routes received by the router interface are processed based on the contents of the static-to-rip route map. Routes that match criteria specified in this route map are either allowed or denied redistribution into the RIPng network. The route map may also specify the modification of route information before the route is redistributed. See [“Using Route Maps” on page 26-15](#) for more information.

To remove a route map redistribution configuration, use the **no** form of the **ipv6 redistrib** command. For example:

```
-> no ipv6 redistrib static into rip route-map static-to-rip
```

Use the **show ipv6 redistrib** command to verify the redistribution configuration:

```
-> show ipv6 redistrib
```

Source Protocol	Destination Protocol	Status	Route Map
localIPv6	RIPng	Enabled	ipv6rm
Static	RIPng	Enabled	static-to-rip

Configuring the Administrative Status of the Route Map Redistribution

The administrative status of a route map redistribution configuration is enabled by default. To change the administrative status, use the **status** parameter with the **ipv6 redistrib** command. For example, the following command disables the redistribution administrative status for the specified route map:

```
-> ipv6 redistrib static into rip route-map static-to-rip status disable
```

The following command example enables the administrative status:

```
-> ipv6 redistrib static into rip route-map static-to-rip status enable
```

Route Map Redistribution Example

The following example configures the redistribution of Static routes into a RIPng network using a route map (static-to-rip) to filter specific routes:

```
-> ip route-map static-to-rip sequence-number 10 action deny
-> ip route-map static-to-rip sequence-number 10 match tag 5

-> ip route-map static-to-rip sequence-number 20 action permit
-> ip route-map static-to-rip sequence-number 20 match ipv6-interface
intf_static
-> ip route-map static-to-rip sequence-number 20 set metric 255

-> ip route-map static-to-rip sequence-number 30 action permit
-> ip route-map static-to-rip sequence-number 30 set tag 8

-> ip redistrib static into rip route-map static-to-rip
```

The resulting static-to-rip route map redistribution configuration does the following:

- Denies the redistribution of routes with a tag set to five.
- Redistributes into RIPng all routes learned on the intf_static interface and sets the metric for such routes to 255.
- Redistributes into RIPng all other routes (those not processed by sequence 10 or 20) and sets the tag for such routes to eight.

Verifying the IPv6 Configuration

A summary of the show commands used for verifying the IPv6 configuration is given here:

show ipv6 rip	Displays the RIPng status and general configuration parameters.
show ipv6 redistrib	Displays the route map redistribution configuration.
show ipv6 interface	Displays the status and configuration of IPv6 interfaces.
show ipv6 routes	Displays the IPv6 Forwarding Table.
show ipv6 route-pref	Displays the configured route preference of a router.
show ipv6 router database	Displays a list of all routes (static and dynamic) that exist in the IPv6 router database.
show ipv6 prefixes	Displays IPv6 subnet prefixes used in router advertisements.
show ipv6 hosts	Displays the IPv6 Local Host Table.
show ipv6 neighbors	Displays the IPv6 Neighbor Table.
show ipv6 traffic	Displays statistics for IPv6 traffic.
show ipv6 icmp statistics	Displays ICMP6 statistics.
show ipv6 pmtu table	Displays the IPv6 Path MTU Table.
show ipv6 tcp ports	Displays TCP Over IPv6 Connection Table. Contains information about existing TCP connections between IPv6 endpoints.
show ipv6 udp ports	Displays the UDP Over IPv6 Listener Table. Contains information about UDP/IPv6 endpoints.

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

27 Configuring RIP

Routing Information Protocol (RIP) is a widely used Interior Gateway Protocol (IGP) that uses hop count as its routing metric. RIP-enabled routers update neighboring routers by transmitting a copy of their own routing table. The RIP routing table uses the most efficient route to a destination, that is, the route with the fewest hops and longest matching prefix.

The switch supports RIP version 1 (RIPv1), RIP version 2 (RIPv2), and RIPv2 that is compatible with RIPv1. It also supports text key and MD5 authentication, on an interface basis, for RIPv2.

In This Chapter

This chapter describes RIP and how to configure it through the Command Line Interface (CLI). It includes instructions for configuring basic RIP routing and fine-tuning RIP by using optional RIP configuration parameters (e.g., RIP send/receive option and RIP interface metric). It also details RIP redistribution. CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

This chapter provides an overview of RIP and includes information about the following procedures:

- RIP Routing
 - Loading RIP (see [page 27-6](#))
 - Enabling RIP (see [page 27-7](#))
 - Creating a RIP Interface (see [page 27-7](#))
 - Enabling a RIP Interface (see [page 27-7](#))
- RIP Options
 - Configuring the RIP Forced Hold-Down Interval (see [page 27-9](#))
 - Configuring the RIP Update Interval (see [page 27-9](#))
 - Configuring the RIP Invalid Timer (see [page 27-10](#))
 - Configuring the RIP Garbage Timer (see [page 27-10](#))
 - Configuring the RIP Hold-Down Timer (see [page 27-10](#))
 - Enabling a RIP Host Route (see [page 27-11](#))
- RIP Redistribution
 - Configuring Route Redistribution (see [page 27-12](#))
- RIP Security
 - Configuring Authentication Type (see [page 27-18](#))
 - Configuring Passwords (see [page 27-18](#))

RIP Specifications

RFCs Supported	RFC 1058–RIP v1 RFC 2453–RIP v2 RFC 1722–RIP v2 Protocol Applicability Statement RFC 1724–RIP v2 MIB Extension
Platforms Supported	OmniSwitch 6250 Series
Maximum Number of RIP Peers	10
Maximum Number of RIP Interfaces	10
Maximum Number of RIP Routes	256

RIP Defaults

The following table lists the defaults for RIP configuration through the **ip rip** command.

Description	Command	Default
RIP Status	ip rip status	disable
RIP Forced Hold-Down Interval	ip rip force-holddowntimer	0
RIP Update Interval	ip rip update-interval	30 seconds
RIP Invalid Timer	ip rip invalid-timer	180 seconds
RIP Garbage Timer	ip rip garbage-timer	120 seconds
RIP Hold-Down Timer	ip rip holddown-timer	0
RIP Interface Metric	ip rip interface metric	1
RIP Interface Send Version	ip rip interface send-version	v2
RIP Interface Receive Version	ip rip interface recv-version	both
RIP Host Route	ip rip host-route	enable
RIP Route Tag	ip rip host-route	0

Quick Steps for Configuring RIP Routing

To forward packets to a device on a different VLAN, you must create a router interface on each VLAN. To route packets by using RIP, you must enable RIP and create a RIP interface on the router interface. The following steps show you how to enable RIP routing between VLANs “from scratch”. If active VLANs and router ports have already been created on the switch, go to Step 7.

- 1 Create VLAN 1 with a description (e.g., VLAN 1) by using the **vlan** command. For example:

```
-> vlan 1 name "VLAN 1"
```

- 2 Create VLAN 2 with a description (e.g., VLAN 2) by using the **vlan** command. For example:

```
-> vlan 2 name "VLAN 2"
```

- 3 Assign an active port to VLAN 1 by using the **vlan port default** command. For example, the following command assigns port 1 on slot 1 to VLAN 1:

```
-> vlan 1 port default 1/1
```

- 4 Assign an active port to VLAN 2 by using the **vlan port default** command. For example, the following command assigns port 2 on slot 1 to VLAN 2:

```
-> vlan 2 port default 1/2
```

- 5 Configure an IP interface to enable IP routing on a VLAN by using the **ip interface** command. For example:

```
-> ip interface vlan-1 address 171.10.1.1 vlan 1
```

- 6 Configure an IP interface to enable IP routing on a VLAN by using the **ip interface** command. For example:

```
-> ip interface vlan-2 address 171.11.1.1 vlan 2
```

- 7 Load RIP into the switch memory by using the **ip load rip** command. For example:

```
-> ip load rip
```

- 8 Enable RIP on the switch by using the **ip rip status** command. For example:

```
-> ip rip status enable
```

- 9 Create a RIP interface on VLAN 1 by using the **ip rip interface** command. For example:

```
-> ip rip interface vlan-1
```

- 10 Enable the RIP interface by using the **ip rip interface status** command. For example:

```
-> ip rip interface vlan-1 status enable
```

- 11 Create an RIP interface on VLAN 2 by using the **ip rip interface** command. For example:

```
-> ip rip interface vlan-2
```

Note. For more information on VLANs and router ports, see [Chapter 4, “Configuring VLANs.”](#)

RIP Overview

In switching, traffic may be transmitted from one media type to another within the same VLAN. Switching happens at Layer 2, the link layer; routing happens at Layer 3, the network layer. In IP routing, traffic can be transmitted across VLANs. When IP routing is enabled, the switch uses routing protocols to build routing tables that keep track of stations in the network and decide the best path for forwarding data. When the switch receives a packet to be routed, it strips off the MAC header and examines the IP header. It looks up the source/destination address in the routing table, and then adds the appropriate MAC address to the packet. Calculating routing tables and stripping/adding MAC headers to packets is performed by switch software.

IP is associated with several Layer 3 routing protocols. RIP is built into the base code loaded onto the switch. Others are part of Alcatel-Lucent's optional Advanced Routing Software. RIP is an IGP that defines how routers exchange information. RIP makes routing decisions by using a "least-cost path" method. RIPv1 and RIPv2 services allow the switch to learn routing information from neighboring RIP routers. For more information and instructions for configuring RIP, see ["RIP Routing" on page 27-6](#).

When RIP is initially enabled on a switch, it issues a request for routing information, and listens for responses to the request. If a switch configured to supply RIP hears the request, it responds with a response packet based on information in its routing database. The response packet contains destination network addresses and the routing metric for each destination. When a RIP response packet is received, RIP takes the information and rebuilds the switch's routing database, adding new routes and "better" (lower metric) routes to destinations already listed in the database.

RIP uses a hop count metric to measure the distance to a destination. In the RIP metric, a switch advertises directly connected networks at a metric of 1. Networks that are reachable through one other gateway are 2 hops, networks that are reachable through two gateways are 3 hops, etc. Thus, the number of hops (or hop count) along a path from a given source to a given destination refers to the number of networks that are traversed by a datagram along that path. When a switch receives a routing update that contains a new or changed destination network entry, the switch adds one to the metric value indicated in the update and enters the network in the routing table. After updating its routing table, the switch immediately begins transmitting routing updates to inform other network switches of the change. These updates are sent independently of the regularly scheduled updates. By default, RIP packets are broadcast every 30 seconds, even if no change has occurred anywhere in a route or service.

RIP deletes routes from the database if the next switch to that destination says the route contains more than 15 hops. In addition, all routes through a gateway are deleted by RIP if no updates are received from that gateway for a specified time period. If a gateway is not heard from for 120 seconds, all routes from that gateway are placed in a hold-down state. If the hold-down timer value is exceeded, the routes are deleted from the routing database. These intervals also apply to deletion of specific routes.

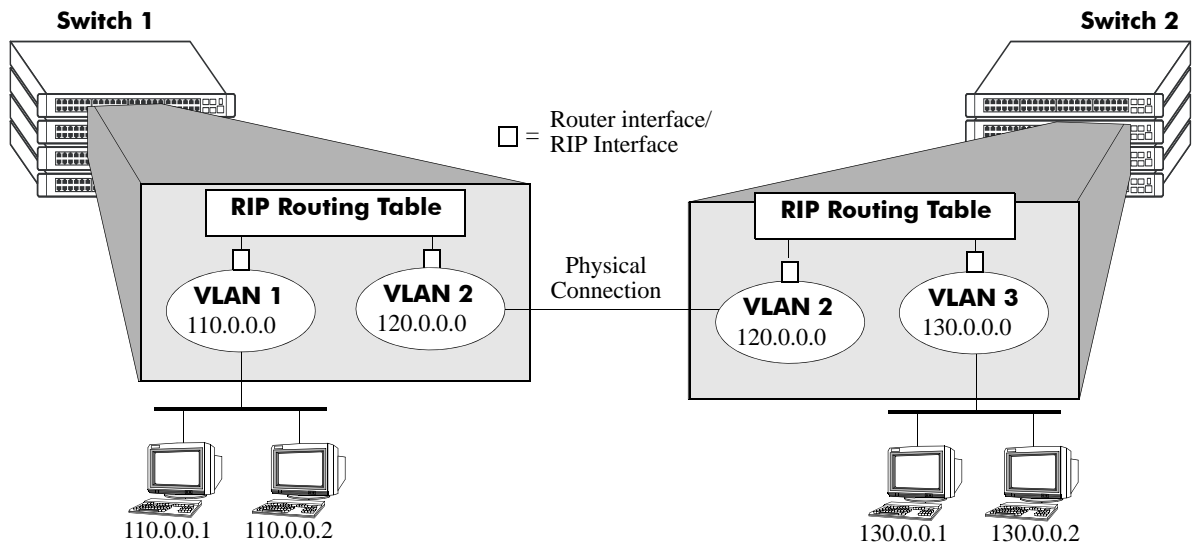
RIP Version 2

RIP version 2 (RIPv2) adds additional capabilities to RIP. Not all RIPv2 enhancements are compatible with RIPv1. To avoid supplying information to RIPv1 routes that could be misinterpreted, RIPv2 can only use non-compatible features when its packets are multicast. Multicast is not supported by RIPv1. On interfaces that are not compatible with IP multicast, the RIPv1-compatible packets used do not contain potentially confusing information. RIPv2 enhancements are listed below.

- **Next Hop**—RIPv2 can advertise a next hop other than the switch supplying the routing update. This capability is useful when advertising a static route to a silent switch not using RIP, since packets passing through the silent switch do not have to cross the network twice.
- **Network Mask**—RIPv1 assumes that all subnetworks of a given network have the same network mask. It uses this assumption to calculate the network masks for all routes received. This assumption prevents subnets with different netmasks from being included in RIP packets. RIPv2 adds the ability to specify the network mask with each network in a packet. Because RIPv1 switches ignore the network mask in RIPv2 packets, their calculation of the network mask could possibly be wrong. For this reason, RIPv1-compatible RIPv2 packets cannot contain networks that would be misinterpreted by RIPv1. These networks must only be provided in native RIPv2 packets that are multicast.
- **Authentication**—RIPv2 packets can contain an authentication key that may be used to verify the validity of the supplied routing data. Authentication may be used in RIPv1-compatible RIPv2 packets, but RIPv1 switches will ignore authentication information. Authentication is a simple password in which an authentication key of up to 16 characters is included in the packet. If this key does not match the configured authentication key, the packet is discarded. For more information on RIP authentication, see [“RIP Security” on page 27-18](#).
- **IP Multicast**—IP Multicast Switching (IPMS) is a one-to-many communication technique employed by emerging applications such as video distribution, news feeds, netcasting, and resource discovery. Unlike unicast, which sends one packet per destination, multicast sends one packet to all devices in any subnetwork that has at least one device requesting the multicast traffic. For more information on IPMS, see [Chapter 36, “Configuring IP Multicast Switching.”](#)

RIP Routing

IP routing requires IP router interfaces to be configured on VLANs and a routing protocol to be enabled and configured on the switch. RIP also requires a RIP interface to be created and enabled on the routing interface. In the illustration below, a router interface and RIP interface have been configured on each VLAN. Therefore, workstations connected to ports on VLAN 1 on Switch 1 can communicate with VLAN 2; workstations connected to ports on VLAN 3 on Switch 2 can communicate with VLAN 2. Also, ports from both switches have been assigned to VLAN 2, and a physical connection has been made between the switches. Therefore, workstations connected to VLAN 1 on Switch 1 can communicate with workstations connected to VLAN 3 on Switch 2.



RIP Routing

Loading RIP

When the switch is initially configured, RIP must be loaded into the switch memory. Use the **ip load rip** command to load RIP.

To remove RIP from the switch memory, you must manually edit the **boot.cfg** file. The **boot.cfg** file is an ASCII text-based file that controls many of the switch parameters. Open the file and delete all references to RIP. You must reboot the switch when this is complete.

Note. In simple networks where only IP forwarding is required, you may not want to use RIP. If you are not using RIP, it is best not to load it to save switch resources.

Enabling RIP

RIP is disabled by default. Use the **ip rip status** command to enable RIP routing on the switch. For example:

```
-> ip rip status enable
```

Use the **ip rip status disable** command to disable RIP routing on the switch. Use the **show ip rip** command to display the current RIP status.

Creating a RIP Interface

You must create a RIP interface on a VLAN's IP router interface to enable RIP routing. Enter the **ip rip interface** command followed by the name of the VLAN router port. For example, to create a RIP interface on a router port with a name of rip-1 you would enter:

```
-> ip rip interface rip-1
```

Use the **no ip rip interface** command to delete a RIP interface. Use the **show ip rip interface** command to display configuration and error information for a RIP interface.

Note. You can create a RIP interface even if an IP router interface has not been configured. However, RIP will not function unless a RIP interface is created and enabled on an IP router interface. See [Chapter 4, "Configuring VLANs,"](#) and [Chapter 25, "Configuring IP,"](#) for more information.

Enabling a RIP Interface

Once you have created a RIP interface, you must enable it to enable RIP routing. Use the **ip rip interface status** command followed by the interface IP address to enable a RIP interface. For example, to enable RIP routing on a RIP interface rip-1 you would enter:

```
-> ip rip interface rip-1 status enable
```

To disable an RIP interface, use the **disable** keyword with the **ip rip interface status** command. For example to disable RIP routing on a RIP interface rip-1 you would enter:

```
-> ip rip interface rip-1 status disable
```

Configuring the RIP Interface Send Option

The RIP Send option defines the type(s) of RIP packets that the interface will send. Using this command will override RIP default behavior. Other devices must be able to interpret the information provided by this command or routing information will not be properly exchanged between the switch and other devices on the network.

Use the **ip rip interface send-version** command to configure an individual RIP interface Send option. Enter the IP address of the RIP interface, and then enter a Send option. For example, to configure a RIP interface rip-1 to send only RIPv1 packets you would enter:

```
-> ip rip interface rip-1 send-version v1
```

The Send options are:

- **v1.** Only RIPv1 packets will be sent by the switch.

- **v2.** Only RIPv2 packets will be sent by the switch.
- **v1compatible.** Only RIPv2 broadcast packets (not multicast) will be sent by the switch.
- **none.** Interface will not forward RIP packets.

The default RIP send option is **v2**.

Use the **show ip rip interface** command to display the current interface send option.

Configuring the RIP Interface Receive Option

The RIP Receive option defines the type(s) of RIP packets that the interface will accept. Using this command will override RIP default behavior. Other devices must be able to interpret the information provided by this command or routing information will not be properly exchanged between the switch and other devices on the network.

Use the **ip rip interface recv-version** command to configure an individual RIP interface Receive option. Enter the IP address of the RIP interface, and then enter a Receive option. For example, to configure RIP interface rip-1 to receive only RIPv1 packets you would enter:

```
-> ip rip interface rip-1 recv-version v1
```

The Receive options are:

- **v1.** Only RIPv1 packets will be received by the switch.
- **v2.** Only RIPv2 packets will be received by the switch.
- **both.** Both RIPv1 and RIPv2 packets will be received by the switch.
- **none.** Interface ignores any RIP packets received.

The default RIP receive option is **both**.

Configuring the RIP Interface Metric

You can set priorities for routes generated by a switch by assigning a metric value to routes generated by that switch's RIP interface. For example, routes generated by a neighboring switch may have a hop count of 1. However, you can lower the priority of routes generated by that switch by increasing the metric value for routes generated by the RIP interface.

Note. When you configure a metric for a RIP interface, this metric cost is added to the metric of the incoming route.

Use the **ip rip interface metric** command to configure the RIP metric or cost for routes generated by a RIP interface. Enter the IP address of the RIP interface as well as a metric value. For example, to set a metric value of 2 for the RIP interface rip-1 you would enter:

```
-> ip rip interface rip-1 metric 2
```

The valid metric range is **1** to **15**. The default is **1**.

Use the **show ip rip interface** command to display the current interface metric.

Configuring the RIP Interface Route Tag

Use the **ip rip route-tag** command to configure a route tag value for routes generated by the RIP interface. This value is used to set priorities for RIP routing. Enter the command and the route tag value. For example, to set a route tag value of 1 you would enter:

```
-> ip rip route-tag 1
```

The valid route tag value range is **1** to **2147483647**. The default is **0**.

Use the **show ip rip** command to display the current route tag value.

RIP Options

The following sections detail procedures for configuring RIP options. RIP must be loaded and enabled on the switch before you can configure any of the RIP configuration options.

Configuring the RIP Forced Hold-Down Interval

The RIP forced hold-down timer value defines an amount of time, in seconds, during which routing information regarding better paths is suppressed. A route enters into a forced hold-down state when an update packet is received that indicates the route is unreachable and when this timer is set to a non-zero value. After this timer has expired and if the value is less than 120 seconds, the route enters a hold-down state for the rest of the period until the remainder of the 120 seconds has also expired. During this time the switch will accept any advertisements for better paths that are received.

Note that the RIP forced hold-down timer is *not* the same as the RIP hold-down timer. The forced hold-down timer defines a separate interval that overlaps the hold-down state. During the forced hold-down timer interval, the switch will not accept *better* routes from other gateways. For more information on RIP hold-down timer, see [“Configuring the RIP Hold-Down Timer” on page 27-10](#).

Use the **ip rip force-holddowntimer** command to configure the interval during which a RIP route remains in a forced hold-down state. Enter the command and the forced hold-down interval value, in seconds. For example, to set a forced hold-down interval value of 10 seconds you would enter:

```
-> ip rip force-holddowntimer 10
```

The valid forced hold-down timer range is **0** to **120**. The default is **0**.

Use the **show ip rip** command to display the current forced hold-down timer value.

Configuring the RIP Update Interval

The RIP update interval defines the time interval, in seconds, when routing updates are sent out. This interval value must be less than or equal to one-third the value of the invalid timer.

Use the **ip rip update-interval** command to configure the interval during which a RIP route remains in an update state. Enter the command and the update interval value, in seconds. For example, to set an update interval value of 45 seconds, you would enter:

```
-> ip rip update-interval 45
```

The valid update interval range is **1** to **120**. The default is **30**.

Configuring the RIP Invalid Timer

The RIP invalid timer value defines the time interval, in seconds, during which a route will remain active in the Routing Information Base (RIB) before it is moved to the invalid state. This timer value must be at least three times the update interval value.

Use the `ip rip invalid-timer` command to configure the time interval that must elapse before an active route becomes invalid. Enter the command and the invalid timer value, in seconds. For example, to set an invalid interval value of 270 seconds you would enter:

```
-> ip rip invalid-timer 270
```

The invalid timer range is **3** to **360**. The default is **180**.

Configuring the RIP Garbage Timer

The RIP garbage timer defines the time interval, in seconds, that must elapse before an expired route is removed from the RIB.

Note that during the garbage interval, the router advertises the route with a metric of INFINITY.

Use the `ip rip garbage-timer` command to configure the time interval after which an expired route is removed from the RIB. Enter the command and the garbage timer value, in seconds. For example, to set a garbage timer value of 180 seconds you would enter:

```
-> ip rip garbage-timer 180
```

The garbage timer range is **0** to **180**. The default is **120**.

Configuring the RIP Hold-Down Timer

The RIP hold-down timer defines the time interval, in seconds, during which a route remains in the hold-down state.

Whenever RIP detects a route with a higher metric than the route in the RIB, the route with the higher metric goes into the hold-down state. The route updates with a metric of INFINITY are excluded.

Use the `ip rip holddown-timer` command to configure the interval during which a RIP route remains in the hold-down state. Enter the command and the hold-down timer value, in seconds. For example, to set a hold-down timer value of 10 seconds you would enter:

```
-> ip rip holddown-timer 10
```

The hold-down timer range is **0** to **120**. The default is **0**.

Reducing the Frequency of RIP Routing Updates

To optimize system performance, you can reduce the frequency of the RIP routing updates by increasing the length of the update, invalid, and garbage timers by about 50% above their default values. For example:

```
-> ip rip update-interval 45
-> ip rip invalid-timer 270
-> ip rip garbage-timer 180
```

Enabling a RIP Host Route

A host route differs from a network route, which is a route to a specific network. This command allows a direct connection to the host without using the RIP table. If a switch is directly attached to a host on a network, use the **ip rip host-route** command to enable a default route to the host. For example:

```
-> ip rip host-route
```

The default is to enable a default host route.

Use the **no ip rip host-route** command to disable the host route. Use the **show ip rip** command to display the current host route status.

Configuring Redistribution

It is possible to configure the RIP protocol to advertise routes learned from other routing protocols into the RIP network. Such a process is referred to as route redistribution and is configured using the **ip redistrib** command.

Redistribution uses route maps to control how external routes are learned and distributed. A route map consists of one or more user-defined statements that can determine which routes are allowed or denied access to the RIP network. In addition a route map may also contain statements that modify route parameters before they are redistributed.

When a route map is created, it is given a name to identify the group of statements that it represents. This name is required by the **ip redistrib** command. Therefore, configuring route redistribution involves the following steps:

- 1 Create a route map, as described in [“Using Route Maps” on page 27-12](#).
- 2 Configure redistribution to apply a route map, as described in [“Configuring Route Map Redistribution” on page 27-16](#).

Using Route Maps

A route map specifies the criteria that are used to control redistribution of routes between protocols. Such criteria is defined by configuring route map statements. There are three different types of statements:

- **Action.** An action statement configures the route map name, sequence number, and whether or not redistribution is permitted or denied based on route map criteria.
- **Match.** A match statement specifies criteria that a route must match. When a match occurs, then the action statement is applied to the route.
- **Set.** A set statement is used to modify route information before the route is redistributed into the receiving protocol. This statement is only applied if all the criteria of the route map is met and the action permits redistribution.

The **ip route-map** command is used to configure route map statements and provides the following **action**, **match**, and **set** parameters:

ip route-map action ...	ip route-map match ...	ip route-map set ...
permit	ip-address	metric
deny	ip-nexthop	tag
	ipv6-address	ip-nexthop
	ipv6-nexthop	ipv6-nexthop
	tag	
	ipv4-interface	
	ipv6-interface	
	metric	

Refer to the “IP Commands” chapter in the *OmniSwitch CLI Reference Guide* for more information about the **ip route-map** command parameters and usage guidelines.

Once a route map is created, it is then applied using the **ip redistrib** command. See [“Configuring Route Map Redistribution” on page 27-16](#) for more information.

Creating a Route Map

When a route map is created, it is given a name (up to 20 characters), a sequence number, and an action (permit or deny). Specifying a sequence number is optional. If a value is not configured, then the number 50 is used by default.

To create a route map, use the **ip route-map** command with the **action** parameter. For example,

```
-> ip route-map static-to-rip sequence-number 10 action permit
```

The above command creates the static-to-rip route map, assigns a **sequence number** of 10 to the route map, and specifies a **permit** action.

To optionally filter routes before redistribution, use the **ip route-map** command with a **match** parameter to configure match criteria for incoming routes. For example,

```
-> ip route-map static-to-rip sequence-number 10 match tag 8
```

The above command configures a match statement for the static-to-rip route map to filter routes based on their tag value. When this route map is applied, only Static routes with a tag value of eight are redistributed into the RIP network. All other routes with a different tag value are dropped.

Note. Configuring match statements is not required. However, if a route map does not contain any match statements and the route map is applied using the **ip redistrib** command, the router redistributes *all* routes into the network of the receiving protocol.

To modify route information before it is redistributed, use the **ip route-map** command with a **set** parameter. For example,

```
-> ip route-map static-to-rip sequence-number 10 set tag 5
```

The above command configures a set statement for the static-to-rip route map that changes the route tag value to five. Because this statement is part of the static-to-rip route map, it is only applied to routes that have an existing tag value equal to eight.

The following is a summary of the commands used in the above examples:

```
-> ip route-map static-to-rip sequence-number 10 action permit
-> ip route-map static-to-rip sequence-number 10 match tag 8
-> ip route-map static-to-rip sequence-number 10 set tag 5
```

To verify a route map configuration, use the **show ip route-map** command:

```
-> show ip route-map
Route Maps: configured: 1 max: 200
Route Map: static-to-rip Sequence Number: 10 Action permit
      match tag 8
      set tag 5
```

Deleting a Route Map

Use the **no** form of the **ip route-map** command to delete an entire route map, a route map sequence, or a specific statement within a sequence.

To delete an entire route map, enter **no ip route-map** followed by the route map name. For example, the following command deletes the entire route map named `redistipv4`:

```
-> no ip route-map redistipv4
```

To delete a specific sequence number within a route map, enter **no ip route-map** followed by the route map name, then **sequence-number** followed by the actual number. For example, the following command deletes sequence 10 from the `redistipv4` route map:

```
-> no ip route-map redistipv4 sequence-number 10
```

Note that in the above example, the `redistipv4` route map is not deleted. Only those statements associated with sequence 10 are removed from the route map.

To delete a specific statement within a route map, enter **no ip route-map** followed by the route map name, then **sequence-number** followed by the sequence number for the statement, then either **match** or **set** and the match or set parameter and value. For example, the following command deletes only the match tag 8 statement from route map `redistipv4` sequence 10:

```
-> no ip route-map redistipv4 sequence-number 10 match tag 8
```

Configuring Route Map Sequences

A route map may consist of one or more sequences of statements. The sequence number determines which statements belong to which sequence and the order in which sequences for the same route map are processed.

To add match and set statements to an existing route map sequence, specify the same route map name and sequence number for each statement. For example, the following series of commands creates route map `rm_1` and configures match and set statements for the `rm_1` sequence 10:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 8
-> ip route-map rm_1 sequence-number 10 set metric 1
```

To configure a new sequence of statements for an existing route map, specify the same route map name but use a different sequence number. For example, the following command creates a new sequence 20 for the `rm_1` route map:

```
-> ip route-map rm_1 sequence-number 20 action permit
-> ip route-map rm_1 sequence-number 20 match ipv4-interface to-finance
-> ip route-map rm_1 sequence-number 20 set metric 5
```

The resulting route map appears as follows:

```
-> show ip route-map rm_1
Route Map: rm_1 Sequence Number: 10 Action permit
  match tag 8
  set metric 1
Route Map: rm_1 Sequence Number: 20 Action permit
  match ipv4 interface to-finance
  set metric 5
```

Sequence 10 and sequence 20 are both linked to route map `rm_1` and are processed in ascending order according to their sequence number value. Note that there is an implied logical OR between sequences. As a result, if there is no match for the tag value in sequence 10, then the match interface statement in sequence 20 is processed. However, if a route matches the tag 8 value, then sequence 20 is not used. The set statement for whichever sequence was matched is applied.

A route map sequence may contain multiple match statements. If these statements are of the same kind (e.g., match tag 5, match tag 8, etc.) then a logical OR is implied between each like statement. If the match statements specify different types of matches (e.g. match tag 5, match ip4 interface to-finance, etc.), then a logical AND is implied between each statement. For example, the following route map sequence will redistribute a route if its tag is either 8 or 5:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 5
-> ip route-map rm_1 sequence-number 10 match tag 8
```

The following route map sequence will redistribute a route if the route has a tag of 8 or 5 *and* the route was learned on the IPv4 interface to-finance:

```
-> ip route-map rm_1 sequence-number 10 action permit
-> ip route-map rm_1 sequence-number 10 match tag 5
-> ip route-map rm_1 sequence-number 10 match tag 8
-> ip route-map rm_1 sequence-number 10 match ipv4-interface to-finance
```

Configuring Access Lists

An IP access list provides a convenient way to add multiple IPv4 or IPv6 addresses to a route map. Using an access list avoids having to enter a separate route map statement for each individual IP address. Instead, a single statement is used that specifies the access list name. The route map is then applied to all the addresses contained within the access list.

Configuring an IP access list involves two steps: creating the access list and adding IP addresses to the list. To create an IP access list, use the **ip access-list** command (IPv4) or the **ipv6 access-list** command (IPv6) and specify a name to associate with the list. For example,

```
-> ip access-list ipaddr
-> ipv6 access-list ip6addr
```

To add addresses to an access list, use the **ip access-list address** (IPv4) or the **ipv6 access-list address** (IPv6) command. For example, the following commands add addresses to an existing access list:

```
-> ip access-list ipaddr address 16.24.2.1/16
-> ipv6 access-list ip6addr address 2001::1/64
```

Use the same access list name each time the above commands are used to add additional addresses to the same access list. In addition, both commands provide the ability to configure if an address and/or its matching subnet routes are permitted (the default) or denied redistribution. For example:

```
-> ip access-list ipaddr address 16.24.2.1/16 action deny redistrib-control all-
subnets
-> ipv6 access-list ip6addr address 2001::1/64 action permit redistrib-control no-
subnets
```

For more information about configuring access list commands, see the “IP Commands” chapter in the *OmniSwitch CLI Reference Guide*.

Configuring Route Map Redistribution

The **ip redistrib** command is used to configure the redistribution of routes from a source protocol into the RIP destination protocol. This command is used on the RIP router that will perform the redistribution.

A source protocol is a protocol from which the routes are learned. A destination protocol is the one into which the routes are redistributed. Make sure that both protocols are loaded and enabled before configuring redistribution.

Redistribution applies criteria specified in a route map to routes received from the source protocol. Therefore, configuring redistribution requires an existing route map. For example, the following command configures the redistribution of Static routes into the RIP network using the static-to-rip route map:

```
-> ip redistrib static into rip route-map static-to-rip
```

RIP routes received by the router interface are processed based on the contents of the static-to-rip route map. Routes that match criteria specified in this route map are either allowed or denied redistribution into the RIP network. The route map may also specify the modification of route information before the route is redistributed. See [“Using Route Maps” on page 27-12](#) for more information.

To remove a route map redistribution configuration, use the **no** form of the **ip redistrib** command. For example:

```
-> no ipv6 redistrib static into rip route-map static-to-rip
```

Use the **show ip redistrib** command to verify the redistribution configuration:

```
-> show ip redistrib
```

Source Protocol	Destination Protocol	Status	Route Map
LOCAL4	RIP	Enabled	rip_1

Configuring the Administrative Status of the Route Map Redistribution

The administrative status of a route map redistribution configuration is enabled by default. To change the administrative status, use the **status** parameter with the **ip redistrib** command. For example, the following command disables the redistribution administrative status for the specified route map:

```
-> ip redistrib static into rip route-map static-to-rip status disable
```

The following command example enables the administrative status:

```
-> ip redistrib static into rip route-map static-to-rip status enable
```

Route Map Redistribution Example

The following example configures the redistribution of Static routes into a RIP network using a route map (static-to-rip) to filter specific routes:

```
-> ip route-map static-to-rip sequence-number 10 action deny
-> ip route-map static-to-rip sequence-number 10 match tag 5

-> ip route-map static-to-rip sequence-number 20 action permit
-> ip route-map static-to-rip sequence-number 20 match ipv4-interface
intf_static
-> ip route-map static-to-rip sequence-number 20 set metric 255

-> ip route-map static-to-rip sequence-number 30 action permit
-> ip route-map static-to-rip sequence-number 30 set tag 8

-> ipv6 redist static into rip route-map static-to-rip
```

The resulting static-to-rip route map redistribution configuration does the following:

- Denies the redistribution of routes with a tag set to five.
- Redistributes into RIP all routes learned on the intf_static interface and sets the metric for such routes to 255.
- Redistributes into RIP all other routes (those not processed by sequence 10 or 20) and sets the tag for such routes to eight.

RIP Security

By default, there is no authentication used for a RIP. However, you can configure a password for a RIP interface. To configure a password, you must first select the authentication type (simple or MD5), and then configure a password.

Configuring Authentication Type

If simple or MD5 password authentication is used, both switches on either end of a link must share the same password. Use the **ip rip interface auth-type** command to configure the authentication type. Enter the name of the RIP interface, and then enter an authentication type:

- **none.** No authentication will be used.
- **simple.** Simple password authentication will be used.
- **md5.** MD5 authentication will be used.

For example, to configure the RIP interface rip-1 for simple authentication you would enter:

```
-> ip rip interface rip-1 auth-type simple
```

To configure the RIP interface rip-1 for MD5 authentication you would enter:

```
-> ip rip interface rip-1 md5 auth-type md5
```

Configuring Passwords

If you configure simple or MD5 authentication you must configure a text string that will be used as the password for the RIP interface. If a password is used, all switches that are intended to communicate with each other must share the same password.

After configuring the interface for simple authentication as described above, configure the password for the interface by using the **ip rip interface auth-key** command. Enter the IP address of the RIP interface, and then enter a 16-byte text string. For example to configure a password “nms” you would enter:

```
-> ip rip interface rip-1 auth-key nms
```

Verifying the RIP Configuration

A summary of the show commands used for verifying the RIP configuration is given here:

show ip rip	Displays the RIP status and general configuration parameters (e.g., forced hold-down timer).
show ip rip routes	Displays the RIP routing database. The routing database contains all the routes learned through RIP.
show ip rip interface	Displays the RIP interface status and configuration.
show ip rip peer	Displays active RIP neighbors (peers).
show ip redistrib	Displays the currently configured RIP redistribution filters.

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

28 Configuring RDP

Router Discovery Protocol (RDP) is an extension of ICMP that allows end hosts to discover routers on their networks. This implementation of RDP supports the router requirements as defined in RFC 1256.

In This Chapter

This chapter describes the RDP feature and how to configure RDP parameters through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

The following procedures are described:

- [“Enabling/Disabling RDP” on page 28-8.](#)
- [“Creating an RDP Interface” on page 28-8.](#)
- [“Specifying an Advertisement Destination Address” on page 28-9.](#)
- [“Defining the Advertisement Interval” on page 28-9.](#)
- [“Setting the Advertisement Lifetime” on page 28-10.](#)
- [“Setting the Preference Levels for Router IP Addresses” on page 28-10.](#)
- [“Verifying the RDP Configuration” on page 28-11.](#)

RDP Specifications

RFCs Supported	RFC 1256–ICMP Router Discovery Messages
Platforms Supported	OmniSwitch 6250 Series
Router advertisements	Supported
Host solicitations	Only responses to solicitations supported.
Maximum number of RDP interfaces per switch	One for each available IP interface configured on the switch.
Advertisement destination addresses	224.0.0.1 (all systems multicast) 255.255.255.255 (broadcast)

RDP Defaults

Parameter Description	CLI Command	Default Value/Comments
RDP status for the switch	ip router-discovery	Disabled
RDP status for switch interfaces (router VLAN IP addresses)	ip router-discovery interface	Disabled
Advertisement destination address for an active RDP interface.	ip router-discovery interface advertisement-address	All systems multicast (224.0.0.1)
Maximum time between advertisements sent from an active RDP interface	ip router-discovery interface max-advertisement-interval	600 seconds
Minimum time between advertisements sent from an active RDP interface	ip router-discovery interface min-advertisement-interval	450 seconds (0.75 * maximum advertisement interval)
Maximum time IP addresses contained in an advertisement packet are considered valid	ip router-discovery interface advertisement-lifetime	1800 seconds (3 * maximum advertisement interval)
Preference level for IP addresses contained in an advertisement packet	ip router-discovery interface preference-level	0

Quick Steps for Configuring RDP

Configuring RDP involves enabling RDP operation on the switch and creating RDP interfaces to advertise VLAN router IP addresses on the LAN. There is no order of configuration involved. For example, it is possible to create RDP interfaces even if RDP is not enabled on the switch.

The following steps provide a quick tutorial on how to configure RDP. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

- 1 Enable RDP operation on the switch.

```
-> ip router-discovery enable
```

Note. *Optional.* To verify the global RDP configuration for the switch, enter the **show ip router-discovery** command. The display is similar to the one shown below:

```
-> show ip router-discovery
Status                = Enabled,
RDP uptime            = 161636 secs
#Packets Tx           = 4,
#Packets Rx           = 0,
#Send Errors          = 0,
#Recv Errors          = 0,
```

For more information about this command, refer to the “RDP Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

- 2 Use the following command to create an RDP interface for an IP router interface. In this example, an RDP interface is created for the IP router interface named Marketing (note that the IP interface is referenced by its name).

```
-> ip router-discovery interface Marketing enable
```

- 3 When an RDP interface is created, default values are set for the interface advertisement destination address, transmission interval, lifetime, and preference level parameters. If you want to change the default values for these parameters, see “Creating an RDP Interface” on page 28-8.

Note. *Optional.* To verify the RDP configuration for all RDP interfaces, enter the **show ip router-discovery interface** command. The display is similar to the one shown below:

```
-> show ip router-discovery interface
      IP i/f   RDP i/f Next   #Pkts
      Name     status   status Advt sent recvd
-----+-----+-----+-----+-----+-----
Marketing      Disabled Enabled   9    0    0
Finance IP Network Disabled Enabled   3    0    0
```

To verify the configuration for a specific RDP interface, specify the interface name when using the **show ip router-discovery interface** command. The display is similar to the one shown below:

```
-> show ip router-discovery interface Marketing
Name = Marketing,
IP Address = 11.255.4.1,
IP Mask = 255.0.0.0,
IP Interface status = Enabled,
RDP Interface status = Enabled,
Advertisement address = 224.0.0.1,
Max Advertisement interval = 600 secs,
Min Advertisement interval = 450 secs,
Advertisement lifetime = 1800 secs,
Preference Level = 0x0,
#Packets sent = 3,
#Packets received = 0
```

For more information about this command, refer to the “RDP Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

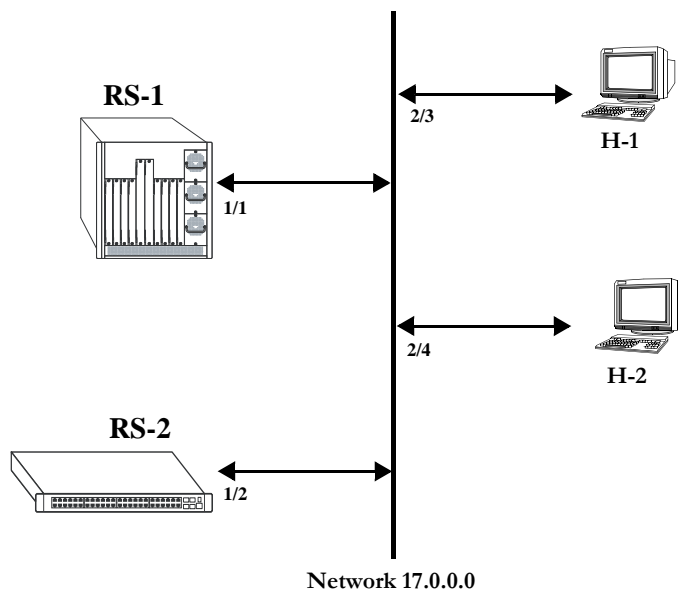
RDP Overview

End host (clients) sending traffic to other networks need to forward their traffic to a router. In order to do this, hosts need to find out if one or more routers exist on their LAN, then learn their IP addresses. One way to discover neighboring routers is to manually configure a list of router IP addresses that the host reads at startup. Another method available involves listening to routing protocol traffic to gather a list of router IP addresses.

RDP provides an alternative method for hosts to discover routers on their network that involves the use of ICMP advertisement and solicitation messages. Using RDP, hosts attached to multicast or broadcast networks send solicitation messages when they start up. Routers respond to solicitation messages with an advertisement message that contains the router IP addresses. In addition, routers first send advertisement messages when their RDP interface becomes active, and then subsequently at random intervals.

When a host receives a router advertisement message, it adds the IP addresses contained in the message to its list of default router gateways in the order of preference. As a result, the list of router IP addresses is dynamically created and maintained, eliminating the need for manual configuration of such a list. In addition, hosts do not have to recognize many different routing protocols to discover router IP addresses.

The following diagram illustrates an example of using RDP in a typical network configuration:



RDP Application Example

When interfaces 2/3 and 2/4 on hosts H-1 and H-2, respectively, become active, they transmit router solicitation ICMP messages on Network 17.0.0.0. The RDP enabled routers RS-1 and RS-2 pick up these packets on their RDP interfaces 1/1 and 1/2 and respond with router advertisement ICMP messages. RS-1 and RS-2 also periodically send out router advertisements on their RDP interfaces.

RDP Interfaces

An RDP interface is created by enabling RDP on a VLAN router IP address. Once enabled, the RDP interface becomes active and joins the all-routers IP multicast group (224.0.0.2). The interface then transmits three initial router advertisement messages at random intervals that are no greater than 16 seconds apart. This process occurs upon activation to increase the likelihood that end hosts will quickly discover this router.

After an RDP interface becomes active and transmits its initial advertisements, subsequent advertisements are transmitted at random intervals that fall between a configurable range of time. This range of time is defined by specifying a maximum and minimum advertisement interval value. See [“Defining the Advertisement Interval” on page 28-9](#) for more information. Because advertisements are transmitted at random intervals, the risk of system overload is reduced as advertisements from other routers on the same link are not likely to transmit at the same time.

It is important to note that advertisements are only transmitted on RDP interfaces if the following conditions are met:

- The RDP global status is enabled on the switch.
- An IP interface exists and is in the enabled state.
- An RDP interface exists and is in the enabled state.

The router advertisement is a multicast packet sent to the all-systems IP multicast group (224.0.0.1) or the broadcast address. Note that RDP is not recommended for detecting neighboring router failures, referred to as black holes, in the network. However, it is possible to use RDP as a supplement for black hole detection by setting RDP interface advertisement interval and lifetime values to values lower than the default values for these parameters. See [“Defining the Advertisement Interval” on page 28-9](#) and [“Setting the Advertisement Lifetime” on page 28-10](#) for more information.

Security Concerns

ICMP RDP packets are not authenticated, which makes them vulnerable to the following attacks:

- **Passive monitoring**—Attackers can use RDP to re-route traffic from vulnerable systems through the attacker's system. This allows the attacker to monitor or record one side of the conversation. However, the attacker must reside on the same network as the victim for this scenario to work.
- **Man in the middle**—Attacker modifies any of the outgoing traffic or plays man in the middle, acting as a proxy between the router and the end host. In this case, the victim thinks that it is communicating with an end host, not an attacker system. The end host thinks that it is communicating with a router because the attacker system is passing information through to the host from the router. If the victim is a secure Web server that uses SSL, the attacker sitting in between the server and an end host could intercept unencrypted traffic. As is the case with passive monitoring, the attacker must reside on the same network as the victim for this scenario to work.
- **Denial of service (DoS)**—Remote attackers can spoof these ICMP packets and remotely add bad default-route entries into a victim's routing table. This would cause the victim to forward frames to the wrong address, thus making it impossible for the victim's traffic to reach other networks. Because of the large number of vulnerable systems and the fact that this attack will penetrate firewalls that do not stop incoming ICMP packets, this DoS attack can become quite severe. (See [Chapter 25, "Configuring IP,"](#) and [Chapter 34, "Configuring QoS,"](#) for more information about DoS attacks.)

Note. Security concerns associated with using RDP are generic to the feature as defined in RFC 1256 and not specific to this implementation.

Enabling/Disabling RDP

RDP is included in the base software and is available when the switch starts up. However, by default this feature is not operational until it is enabled on the switch.

To enable RDP operation on the switch, use the following command:

```
-> ip router-discovery enable
```

Once enabled, any existing RDP interfaces on the switch that are also enabled will activate and start to send initial advertisements. See [“RDP Interfaces” on page 28-6](#) for more information.

To disable RDP operation on the switch, use the following command:

```
-> ip router-discovery disable
```

Use the [show ip router-discovery](#) command to determine the current operational status of RDP on the switch.

Creating an RDP Interface

An RDP interface is created by enabling RDP for an existing IP router interface, which is then advertised by RDP as an active router on the local network. Note that an RDP interface is not active unless RDP is also enabled for the switch.

To create an RDP interface, enter **ip router-discovery interface** followed by the name of the IP router interface, and then **enable**. For example, the following command creates an RDP interface for the IP router interface named Marketing:

```
-> ip router-discovery interface Marketing enable
```

The IP router interface name is the name assigned to the interface when it was first created. For more information about creating IP router interfaces, see [Chapter 25, “Configuring IP.”](#)

The first time an RDP interface is enabled, it is not necessary to enter **enable** as part of the command. However, if the interface is subsequently disabled, then entering **enable** is required the next time this command is used. For example, the following sequence of commands initially enables an RDP interface for the Marketing IP router interface, then disables and again enables the same interface:

```
-> ip router-discovery interface Marketing
-> ip router-discovery interface Marketing disable
-> ip router-discovery interface Marketing enable
```

When the above RDP interface becomes active, advertisement packets are transmitted on all active ports that belong to the VLAN associated with the Marketing interface. These packets contain the IP address associated with the Marketing interface for the purposes of advertising this interface on the network.

When an RDP interface is created, it is automatically configured with the following default parameter values:

RDP Interface Parameter	Default
Advertisement destination address.	All systems multicast (224.0.0.1)
Advertisement time interval defined by maximum and minimum values.	Maximum = 600 seconds Minimum = 450 seconds (0.75 * maximum value)

RDP Interface Parameter	Default
Advertisement lifetime.	1800 seconds (3 * maximum value)
Router IP address preference level.	0

It is only necessary to change the above parameter values if the default value is not sufficient. The following subsections provide information about how to configure RDP interface parameters if it is necessary to use a different value.

Specifying an Advertisement Destination Address

Active RDP interfaces transmit advertisement packets at random intervals and in response to ICMP solicitation messages received from network hosts. These packets are sent to one of two supported destination addresses, all systems multicast (224.0.0.1) or broadcast (255.255.255.255).

By default, RDP interfaces are configured to use the 224.0.0.1 as the destination address. To change the RDP destination address, use the [ip router-discovery interface advertisement-address](#) command.

For example, the following command changes the destination address to the broadcast address:

```
-> ip router-discovery interface Marketing advertisement-address broadcast
```

Enter **all-systems-multicast** when using this command to change the destination address to 224.0.0.1. For example:

```
-> ip router-discovery interface Marketing advertisement-address all-systems-  
multicast
```

Defining the Advertisement Interval

The advertisement interval represents a range of time, in seconds, in which the RDP will transmit advertisement packets at random intervals. This range is defined by configuring a maximum amount of time that the RDP will not exceed before the next transmission and configuring a minimum amount of time that the RDP will observe before sending the next transmission. Both of these values are referred to as the maximum advertisement interval and the minimum advertisement interval.

Note that when an RDP interface becomes active, it transmits 3 advertisement packets at intervals no greater than 16 seconds. This facilitates a quick discovery of this router on the network. After these initial transmissions, advertisements occur at random times within the advertisement interval value or in response to solicitation messages received from network hosts.

Setting the Maximum Advertisement Interval

To set the maximum amount of time, in seconds, that the RDP will allow between advertisements, use the [ip router-discovery interface max-advertisement-interval](#) command. For example, the following command sets this value to 1500 seconds for the Marketing IP router interface:

```
-> ip router-discovery interface Marketing max-advertisement-interval 1500
```

Make sure that the value specified with this command is *greater* than the current minimum advertisement interval value. By default, this value is set to 600 seconds.

Setting the Minimum Advertisement Interval

To set the minimum amount of time, in seconds, that the RDP will allow between advertisements, use the **ip router-discovery interface min-advertisement-interval** command. For example, the following command sets this value to 500 seconds for the Marketing IP router interface:

```
-> ip router-discovery interface Marketing min-advertisement-interval 500
```

Make sure that the value specified with this command is *less* than the current maximum advertisement interval value. By default, this value is set to 0.75 * the default maximum interval value (450 seconds if the maximum interval is set to its default value of 600 seconds).

Setting the Advertisement Lifetime

The advertisement lifetime value indicates how long, in seconds, the router IP address contained in an advertisement packet is considered valid by a host. This value is entered into the lifetime field of an advertisement packet so that it is available to hosts that receive these types of packets.

If a host does not receive another packet from the same router before the lifetime value expires, it assumes the router is no longer available and will drop the router IP address from its table. As a result, it is important that the lifetime value is always *greater* than the current maximum advertisement interval to ensure router transmissions occur before the lifetime value expires.

To set the advertisement lifetime value for packets transmitted from a specific RDP interface, use the **ip router-discovery interface advertisement-lifetime** command. For example, the following command sets this value to 3000 seconds for RDP packets sent from the Marketing IP router interface:

```
-> ip router-discovery interface Marketing advertisement-lifetime 3000
```

By default, the lifetime value is set to 3 * the current maximum interval value (1800 seconds if the maximum interval is set to its default value of 600 seconds).

Setting the Preference Levels for Router IP Addresses

A preference level is assigned to each router IP address contained within an advertisement packet. Hosts will select the IP address with this highest preference level to use as the default router gateway address. By default, this value is set to zero.

To specify a preference level for IP addresses advertised from a specific RDP interface, use the **ip router-discovery interface preference-level** command. For example, the following command sets this value to 10 for the IP address associated with the Marketing IP router interface:

```
-> ip router-discovery interface Marketing preference-level 10
```

Note that router IP address preference levels are only compared with the preference levels of other routers that exist on the same subnet. Set low preference levels to discourage selection of a specific router.

Verifying the RDP Configuration

To display information about the RDP configuration on the switch, use the **show** commands listed below:

show ip router-discovery

Displays the current operational status of RDP on the switch. Also includes the number of advertisement packets transmitted and the number of solicitation packets received by all RDP interfaces on the switch.

show ip router-discovery interface

Displays the current RDP status, related parameter values, and RDP traffic statistics for one or more switch router RDP interfaces.

For more information about the resulting displays from these commands, see the *OmniSwitch 6250 CLI Reference Guide*. An example of the output for the **show ip router-discovery** and **show ip router-discovery interface** commands is also given in [“Quick Steps for Configuring RDP” on page 28-3](#).

29 Configuring DHCP

The User Datagram Protocol (UDP) is a connectionless transport protocol that runs on top of IP networks. The DHCP Relay allows you to use nonroutable protocols (such as UDP) in a routing environment. UDP is used for applications that do not require the establishment of a session and end-to-end error checking. Email and file transfer are two applications that could use UDP. UDP offers a direct way to send and receive datagrams over an IP network and is primarily used for broadcasting messages. This chapter describes the DHCP Relay feature. This feature allows UDP broadcast packets to be forwarded across VLANs that have IP routing enabled.

In This Chapter

This chapter describes the basic components of DHCP Relay and how to configure them. CLI commands are used in the configuration examples. For more details about the syntax of commands, see the *OmniSwitch CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Quick steps for configuring DHCP Relay on [page 29-4](#).
- Setting the IP address for Global DHCP on [page 29-9](#).
- Identifying the VLAN for Per-VLAN DHCP on [page 29-9](#).
- Enabling BOOTP/DHCP Relay on [page 29-10](#).
- Setting the Forward Delay time on [page 29-10](#).
- Setting the Maximum Hops value on [page 29-11](#).
- Setting the Relay Forwarding Option to Standard or Per-VLAN on [page 29-11](#).
- Configuring the DHCP Client Interface to obtain an IP address for the switch on [page 29-12](#).
- Configuring relay for generic UDP service ports on [page 29-14](#).
- Using the Relay Agent Information Option (Option-82) on [page 29-16](#).
- Using DHCP Snooping on [page 29-19](#).

For information about the IP protocol, see [Chapter 25, “Configuring IP.”](#)

DHCP Relay Specifications

RFCs Supported	0951–Bootstrap Protocol 1534–Interoperation between DHCP and BOOTP 1541–Dynamic Host Configuration Protocol 1542–Clarifications and Extensions for the Bootstrap Protocol 2132–DHCP Options and BOOTP Vendor Extensions 3046–DHCP Relay Agent Information Option, 2001 2131–DHCP Client
Platforms Supported	OmniSwitch 6250 Series
DHCP Relay Implementation	Global DHCP Per-VLAN DHCP
DHCP Relay Service	BOOTP/DHCP (Bootstrap Protocol/Dynamic Host Configuration Protocol)
UDP Port Numbers	67 for Request 68 for Response
IP address allocation mechanisms	Dynamic –DHCP assigns an IP address to a host for a limited period of time (or until the host explicitly relinquishes the address). Manual –The network administrator assigns a host’s IP address and the DHCP conveys the address assigned by the host.
IP addresses supported for each Relay Service	Maximum of 256 IP addresses for each Relay Service.
IP addresses supported for the Per-VLAN service	Maximum of 8 IP addresses for each VLAN relay service. Maximum of 256 VLAN relay services.
Maximum number of UDP relay services allowed per switch	12
Maximum number of VLANs to which forwarded UDP service port traffic is allowed	256
Maximum number of DHCP Client interfaces	1 (DHCP Client supported on OS6250-Metro Models Only)
Maximum number of DHCP Snooping VLANs	64
Maximum number of clients per switching ASIC when IP source filtering is enabled.	256

DHCP Relay Defaults

The following table describes the default values of the DHCP Relay parameters:

Parameter Description	Command	Default Value/Comments
Default UDP service	ip udp relay	BOOTP/DHCP
Forward delay time value for DHCP Relay	ip helper forward delay	3 seconds
Maximum number of hops	ip helper maximum hops	4 hops
Packet forwarding option	ip helper standard ip helper per-vlan only	Standard
Automatic switch IP configuration for default VLAN 1	ip helper boot-up	Disabled
Relay Agent Information Option	ip helper agent-informa- tion	Disabled
Switch-level DHCP Snooping	ip helper dhcp-snooping	Disabled
VLAN-level DHCP Snooping	ip helper dhcp-snooping vlan	Disabled

Quick Steps for Setting Up DHCP Relay

You should configure DHCP Relay on switches where packets are routed between IP networks.

There is no separate command for enabling or disabling the relay service. DHCP Relay is automatically enabled on the switch whenever a DHCP server IP address is defined. To set up DHCP Relay, proceed as follows:

1 Identify the IP address of the DHCP server. Where the DHCP server has IP address 128.100.16.1, use the following command:

```
-> ip helper address 128.100.16.1
```

2 Set the forward delay timer for the BOOTP/DHCP relay. To set the timer for a 15 second delay, use the following command:

```
-> ip helper forward delay 15
```

3 Set the maximum hop count value. To set a hop count of 3, use the following command:

```
-> ip helper maximum hops 3
```

Note. Optional. To verify the DHCP Relay configuration, enter the **show ip helper** command. The display shown for the DHCP Relay configured in the above Quick Steps is shown here:

```
-> show ip helper
Forward Delay (seconds) = 15
Max number of hops      = 3
Forward option          = standard
Forwarding Address:
128.100.16.1
```

For more information about this display, see the “DHCP Relay” chapter in the *OmniSwitch CLI Reference Guide*.

DHCP Relay Overview

The DHCP Relay service, its corresponding port numbers, and configurable options are as follows:

- DHCP Relay Service: BOOTP/DHCP
- UDP Port Numbers 67/68 for Request/Response
- Configurable options: DHCP server IP address, Forward Delay, Maximum Hops, Forwarding Option

The port numbers indicate the destination port numbers in the UDP header. The DHCP Relay will verify that the forward delay time (specified by the user) has elapsed before sending the packet down to UDP with the destination IP address replaced by the address (also specified by the user).

If the relay is configured with multiple IP addresses, then the packet will be sent to all IP address destinations. The DHCP Relay also verifies that the maximum hop count has not been exceeded. If the forward delay time is *not* met or the maximum hop count is exceeded, the BOOTP/DHCP packet will be discarded by the DHCP Relay.

The forwarding option allows you to specify if the relay should operate in the standard or per-VLAN only mode. The standard mode forwards all DHCP packets on a global relay service. The per-VLAN only mode forwards DHCP packets that originate from a specific VLAN. See [“Setting the Relay Forwarding Option” on page 29-11](#) for more information.

Alternately the relay function may be provided by an external router connected to the switch; in this case, the relay would be configured on the external router.

DHCP

DHCP (Dynamic Host Configuration Protocol) provides a framework for passing configuration information to Internet hosts on a TCP/IP network. It is based on the Bootstrap Protocol (BOOTP), adding the ability to automatically allocate reusable network addresses and additional configuration options. DHCP consists of the following two components:

- A protocol for delivering host-specific configuration parameters from a DHCP server to a host.
- A mechanism for allocating network addresses to hosts.

DHCP is built on a client-server model in which a designated DHCP server allocates network addresses and delivers configuration parameters to dynamically configured hosts. It supports the following three mechanisms for IP address allocation.

Dynamic—DHCP assigns an IP address to a host for a limited period of time (or until the host explicitly relinquishes the address).

Manual—The network administrator assigns a host's IP address and DHCP simply conveys the assigned address to the host.

DHCP and the OmniSwitch

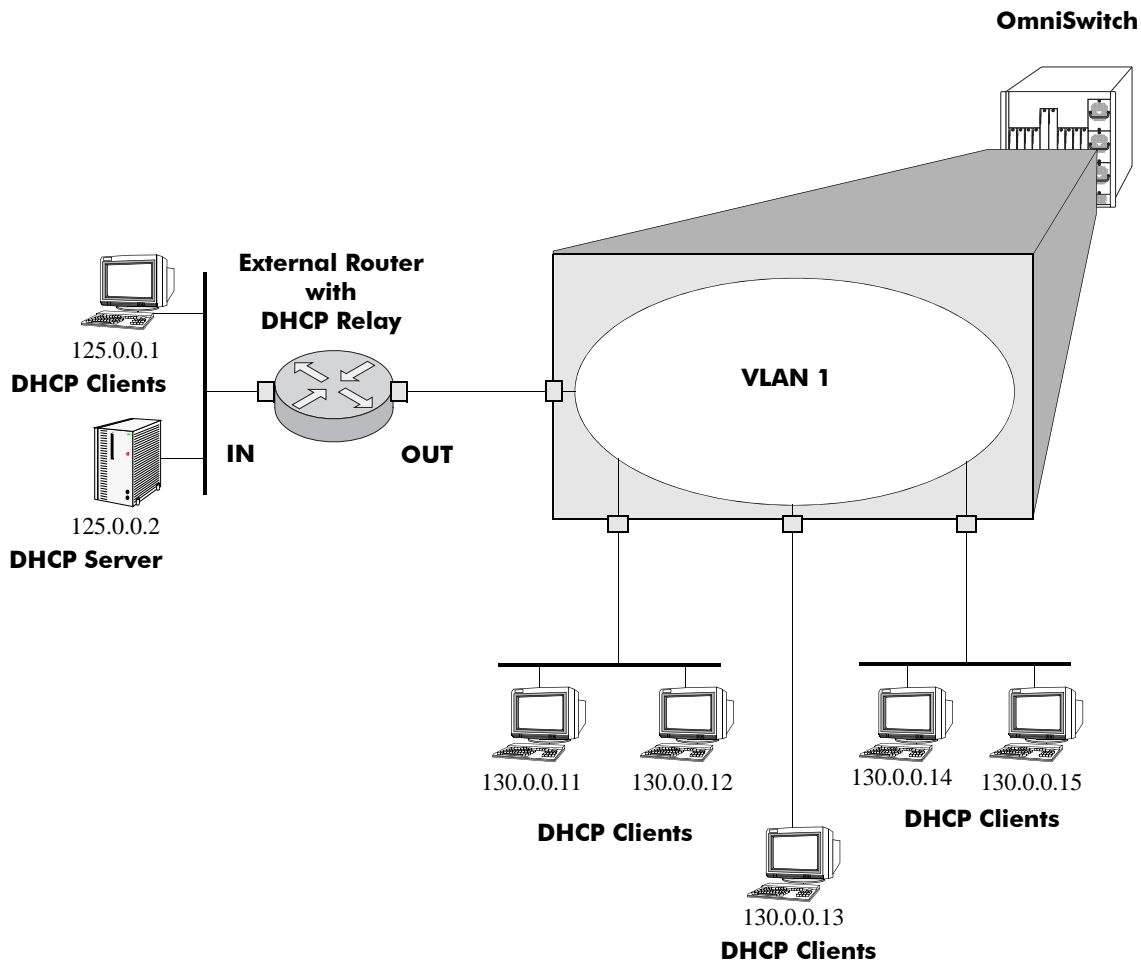
The unique characteristics of the DHCP protocol require a good plan before setting up the switch in a DHCP environment. Since DHCP clients initially have no IP address, placement of these clients in a VLAN is hard to determine. In simple networks (e.g., one VLAN) rules do not need to be deployed to support the BOOTP/DHCP relay functionality.

In multiple VLAN network configurations, VLAN rules can be deployed to strategically support the processing and relay of DHCP packets. The most commonly used rules for this function are IP protocol rules, IP network address rules, and DHCP rules. All of these classify packets received on mobile ports based on the packet protocol type, source IP address, or if the packet is a DHCP request. See [Chapter 8, "Defining VLAN Rules,"](#) for more information.

External DHCP Relay Application

The DHCP Relay may be configured on a router that is external to the switch. In this application example the switched network has a single VLAN configured with multiple segments. All of the network hosts are DHCP-ready, meaning they obtain their network address from the DHCP server. The DHCP server resides behind an external network router, which supports the DHCP Relay functionality.

One requirement for routing DHCP frames is that the router must support DHCP Relay functionality to be able to forward DHCP frames. In this example, DHCP Relay is supported within an external router, which forwards request frames from the incoming router port to the outgoing router port attached to the OmniSwitch.



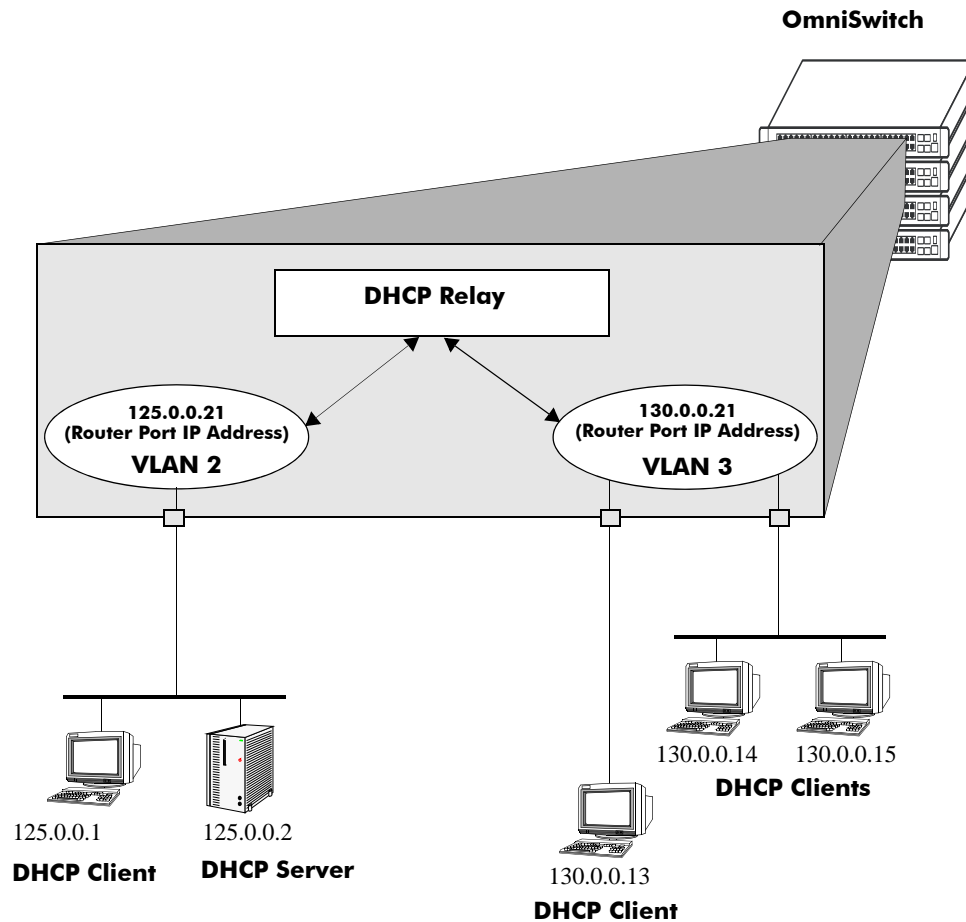
DHCP Clients are Members of the Same VLAN

The external router inserts the subnet address of the first hop segment into the DHCP request frames from the DHCP clients. This subnet address allows the DHCP server to locate the segment on which the requesting client resides. In this example, all clients attached to the OmniSwitch are DHCP-ready and will have the same subnet address (130.0.0.0) inserted into each of the requests by the router's DHCP Relay function. The DHCP server will assign a different IP address to each of the clients. The switch does not need an IP address assigned and all DHCP clients will be members of either a default VLAN or an IP protocol VLAN.

Internal DHCP Relay

The internal DHCP Relay is configured using the UDP forwarding feature in the switch, available through the **ip helper address** command. For more information, see “[DHCP Relay Implementation](#)” on page 29-9.

This application example shows a network with two VLANs, each with multiple segments. All network clients are DHCP-ready and the DHCP server resides on just one of the VLANs. This example is much like the first application example, except that the DHCP Relay function is configured inside the switch.



DHCP Clients in Two VLANs

During initialization, each network client forwards a DHCP request frame to the DHCP server using the local broadcast address. For those locally attached stations, the frame will simply be switched.

In this case, the DHCP server and clients must be members of the same VLAN (they could also all be members of the default VLAN). One way to accomplish this is to use DHCP rules in combination with IP protocol rules to place all IP frames in the same VLAN. See [Chapter 8, “Defining VLAN Rules,”](#) for more information.

Because the clients in the application example are not members of the same VLAN as the DHCP server, they must request an IP address via the DHCP Relay routing entity in the switch. When a DHCP request frame is received by the DHCP Relay entity, it will be forwarded from VLAN 3 to VLAN 2. All the DHCP-ready clients in VLAN 3 must be members of the same VLAN, and the switch must have the DHCP Relay function configured.

DHCP Relay Implementation

The OmniSwitch allows you to configure the DHCP Relay feature in one of two ways. You can set up a global DHCP request or you can set up the DHCP Relay based on the VLAN of the DHCP request. Both of these choices provide the same configuration options and capabilities. However, they are mutually exclusive. The following matrix summarizes the options.

Per-VLAN DHCP Relay	Global DHCP Relay	Effect
Disabled	Disabled	DHCP Request is flooded within its VLAN
Disabled	Enabled	DHCP Request is relayed to the Global Relay
Enabled	Disabled	DHCP Request is relayed to the Per-VLAN Relay
Enabled	Enabled	N/A

Global DHCP

For the global DHCP service, you must identify an IP address for the DHCP server.

Setting the IP Address

The DHCP Relay is automatically enabled on a switch whenever a DHCP server IP address is defined by using the **ip helper address** command. There is no separate command for enabling or disabling the relay service. You should configure DHCP Relay on switches where packets are routed between IP networks. The following command defines a DHCP server address:

```
-> ip helper address 125.255.17.11
```

The DHCP Relay forwards BOOTP/DHCP broadcasts to and from the specified address. If multiple DHCP servers are used, one IP address must be configured for each server. You can configure up to 256 addresses for each relay service.

To delete an IP address, use the **no** form of the **ip helper address** command. The IP address specified with this syntax will be deleted. If an IP address is not specified with this syntax, then *all* IP helper addresses are deleted. The following command deletes an IP helper address:

```
-> ip helper no address 125.255.17.11
```

Per-VLAN DHCP

For the Per-VLAN DHCP service, you must identify the number of the VLAN that makes the relay request.

Identifying the VLAN

You may enter one or more server IP addresses to which packets will be sent from a specified VLAN. Do this by using the **ip helper address vlan** command. The following syntax will identify the IP address 125.255.17.11 as the DHCP server for VLAN 3:

```
-> ip helper address 125.255.17.11 vlan 3
```

The following syntax identifies two DHCP servers for VLAN 4 at two different IP addresses:

```
-> ip helper address 125.255.17.11 125.255.18.11 vlan 4
```

To delete an IP address, use the **no** form of the **ip helper address** command. The IP address specified with this syntax will be deleted. If an IP address is not specified with this syntax, then *all* IP helper addresses are deleted. The following command deletes a helper address for IP address 125.255.17.11:

```
-> ip helper no address 125.255.17.11
```

The following command deletes all IP helper addresses:

```
-> ip helper no address
```

Configuring BOOTP/DHCP Relay Parameters

Once the IP address of the DHCP server(s) is defined and the DHCP Relay is configured for either Global DHCP request or Per-VLAN DHCP request, you can set the following optional parameter values to configure BOOTP relay.

- The forward delay time.
- The hop count.
- The relay forwarding option.

The only parameter that is required for BOOTP relay is the IP address to the DHCP server or to the next hop to the DHCP server. The default values can be accepted for forward delay, hop count, and relay forwarding option.

Alternately the relay function may be provided by an external router connected to the switch; in this case, the relay would be configured on the external router.

Setting the Forward Delay

Forward Delay is a time period that gives the local server a chance to respond to a client before the relay forwards it further out in the network.

The UDP packet that the client sends contains the elapsed boot time. This is the amount of time, measured in seconds, since the client last booted. DHCP Relay will not process the packet unless the client's elapsed boot time value is equal to or greater than the configured value of the forward delay time. If a packet contains an elapsed boot time value that is less than the specified forward delay time value, DHCP Relay discards the packet.

The forward delay time value applies to all defined IP helper addresses. The following command sets the forward delay value of 10 seconds:

```
-> ip helper forward delay 10
```

The range for the forward delay time value is 0 to 65535 seconds.

Setting Maximum Hops

This value specifies the maximum number of relays the BOOTP/DHCP packet can go through until it reaches its server destination. This limit keeps packets from “looping” through the network. If a UDP packet contains a hop count equal to the hops value, DHCP Relay discards the packet. The following syntax is used to set a maximum of four hops:

```
-> ip helper maximum hops 4
```

The hops value represents the maximum number of relays. The range is from one to 16 hops. The default maximum hops value is set to four. This maximum hops value only applies to DHCP Relay. All other switch services will ignore this value.

Setting the Relay Forwarding Option

This value specifies if DHCP Relay should operate in a Standard or Per-VLAN only forwarding mode. By default, the forwarding option is set to standard. To change the forwarding option value, enter **ip helper** followed by **standard** or **per-vlan only**. For example:

```
-> ip helper standard  
-> ip helper per-vlan only
```

Configuring the DHCP Client Interface

The OmniSwitch can be configured with a DHCP Client interface that allows the switch to dynamically obtain an IP address from a DHCP server.

- The DHCP Client interface is configurable on any one VLAN in any VRF instance.
- The DHCP Client interface supports the release and renew functionality according to RFC-2131.
- The option-60 string can be configured on the OmniSwitch and sent as part of the DHCP discover/request packet.
- DHCP option-12 is supported for configuring the OmniSwitch system name.

Configuring the DHCP Client Interface

To enable the DHCP Client functionality use the **ip interface dhcp-client** command. For example:

```
-> ip interface dhcp-client vlan 99
```

When the switch receives a valid IP address lease from a DHCP server:

- The IP address and the subnet mask (DHCP option 1) are assigned to the DHCP Client IP interface.
- A default static route is created according to DHCP option 3 (Router IP Address).
- The lease is periodically renewed and rebound according to the renew time (DHCP option 58) and rebind time (DHCP option 59) returned by the DHCP Server. If the lease cannot be renewed within the lease time (DHCP option 51) returned by the DHCP Server, the IP address will be released. When not specified by the DHCP Server, a default lease time of 7 days is allocated.
- The system name of the OmniSwitch is set according to the hostname (DHCP option 12) returned by the DHCP Server. However, this applies only when the default system name has not already been configured on the OmniSwitch.
- The DHCP Client-enabled IP address serves as the primary IP address when multiple addresses are configured for a VLAN.

Reload and Takeover

The **dhcpClient.db** file is used during a switch reload or CMM takeover to help retain the DHCP server assigned IP address. The IP address saved in this file is the address requested from the DHCP server in the event of a reload or takeover. The following information is stored in the **dhcpClient.db** file located in the */flash/switch* directory on the switch:

- DHCP server assigned IP
- VLAN information
- Subnet mask
- Router IP address
- Checksum value (validates the integrity of the file).

Whenever there is any change in the DHCP server assigned IP address, the **dhcpClient.db** file is updated with the new information and synchronized to the secondary CMM. This file is also synchronized periodically with the DHCP snooping binding table.

The following occurs after a switch reload or takeover:

- The DHCP client interface uses the **dhcpClient.db** file information to create the IP interface with a lease time of 10 minutes and tries to acquire the same IP address.
- After successful renewal of the IP address, the lease time is modified as per the DHCP server assigned IP address.
- If the DHCP client is not able acquire the same IP address, the client will then try to get a new IP address after the switch-assigned DHCP lease time expires. Note that a trap message is sent whenever there is any change to the IP address.

DHCP Client Interface Guidelines

Consider the following when configuring the DHCP Client interface:

- DHCP Client Interface is only supported on OS6250-Metro Models
- The IP address of a DHCP-Client interface is not configurable; this address is assigned only through the DHCP Client process of requesting an IP address.
- DHCP Client only supports IPv4 addresses.
- When using this feature in a stack configuration, enable MAC Retention to ensure that the same IP address is obtained from the DHCP server after takeover.
- Do not configure the DHCP client interface on a switch where the interface will be the relay agent for the client VLAN.
- Although a DHCP Client is configurable for any VLAN in any VRF instance, only one DHCP Client per switch is allowed.
- Make sure the DHCP server is reachable through the DHCP Client VLAN.
- When a DHCP release is performed or the DHCP client interface is deleted, any default static route added for the client is also removed and the corresponding timers (such as release/renew timer) are cancelled.
- When a DHCP release is performed the system name will remain unchanged even if the name was updated using the DHCP client option-12 information.

Configuring UDP Port Relay

In addition to configuring a relay operation for BOOTP/DHCP traffic on the switch, it is also possible to configure relay for generic UDP service ports (i.e., NBNS/NBDD, other well-known UDP service ports, and service ports that are not well-known). This is done using UDP Port Relay commands to enable relay on these types of ports and to specify up to 256 VLANs that can forward traffic destined for these ports.

The UDP Port Relay function is separate from the previously described functions (such as global DHCP, per-VLAN DHCP, and automatic IP configuration) in that using UDP Port Relay does not exclude or prevent other DHCP Relay functionality. However, the following information is important to remember when configuring BOOTP/DHCP relay and UDP port relay:

- UDP port relay supports up to three UDP relay services at any one time and in any combination.

Note. If the relay service for BOOTP/DHCP is disabled when the switch reboots, the service is automatically enabled when the switch comes back up. If there were three non-BOOTP/DHCP relay services already enabled before the reboot, the most recent service enabled is disabled and replaced with the BOOTP/DHCP relay service.

- The **ip helper** commands are used to configure BOOTP/DHCP relay and the **ip udp port** commands are used to configure UDP port relay. The **ip udp relay** command, however, is also used to enable or disable relay for BOOTP/DHCP well known ports 67 and 68.
- If the BOOTP/DHCP relay service is disabled, the **ip helper** configuration is *not* retained and all dependant functionality (i.e., automatic IP configuration for VLAN 1) is disrupted.
- Relaying BOOTP/DHCP traffic is available on a global and per-VLAN basis. Using this function on a per-VLAN basis requires setting the DHCP relay forwarding mode to **per-vlan only**. UDP port relay for generic services is only available on a per-VLAN basis, but does not require enabling the **per-vlan only** forwarding option.

Configuring UDP Port Relay for generic UDP services is a two-step process. The first step involves enabling UDP Port Relay on the generic service port. The second step involves specifying a VLAN that relay will forward traffic destined for the generic service port. Both steps are required and are described below.

Enabling/Disabling UDP Port Relay

By default, a global relay operation is enabled for BOOTP/DHCP relay well-known ports 67 and 68, which becomes active when an IP network host address for a DHCP server is specified. To enable or disable a relay operation for a UDP service port, use the **ip udp relay** command. For example, the following command enables relay on the DNS well-known service port:

```
-> ip udp relay DNS
```

To enable relay on a user-defined (not well-known) UDP service port, then enter the service port number instead of the service name. For example, the following command enables relay on service port 3047:

```
-> ip udp relay 3047
```

To disable a relay operation for a UDP service port, use the **no** form of the **ip udp relay** command. For example, the following command disables relay on the DNS well-known service port:

```
-> no ip udp relay dns
```

For more information about using the **ip udp relay** command, see the *OmniSwitch 6250 CLI Reference Guide*.

Specifying a Forwarding VLAN

To specify which VLAN(s) UDP Port Relay will forward traffic destined for a generic UDP service port, use the **ip udp relay vlan** command. For example, the following command assigns VLAN 5 as a forwarding VLAN for the DNS well-known service port:

```
-> ip udp relay dns vlan 5
```

Note that the **ip udp relay vlan** command only works if UDP Port Relay is already enabled on the specified service port. In addition, when assigning a VLAN to the BOOTP/DHCP service ports, set the DHCP relay forwarding mode to **per-vlan only** first before trying to assign the VLAN.

It is also possible to assign up to 256 forwarding VLANs to each generic service port. To specify more than one VLAN with a single command, enter a range of VLANs. For example, the following command assigns VLANs 6 through 8 and VLAN 10 as forwarding VLANs for the NBNS/NBDD well-known service ports:

```
-> ip udp relay nbnsnbdd vlan 6-8 10
```

If UDP Port Relay was enabled on a not well-known service port, then enter the service port number instead of the service name. For example, the following command assigns VLAN 100 as a forwarding VLAN for UDP service port 3047:

```
-> ip udp relay 3047 vlan 100
```

To remove a VLAN association with a UDP service port, use the **no** form of the **ip udp relay vlan** command. For example, the following command removes the VLAN 6 association with the NBNS/NBDD well-known service port:

```
-> no ip udp relay nbnsnbdd vlan 6
```

For more information about using the **ip udp relay vlan** command, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuring DHCP Security Features

There are two DHCP security features available: DHCP relay agent information option (Option-82) and DHCP Snooping. The DHCP Option-82 feature enables the relay agent to insert identifying information into client-originated DHCP packets before the packets are forwarded to the DHCP server. The DHCP Snooping feature filters DHCP packets between untrusted sources and a trusted DHCP server and builds a binding database to log DHCP client information.

Although DHCP Option-82 is a subcomponent of DHCP Snooping, these two features are mutually exclusive. If the DHCP Option-82 feature is enabled for the switch, then DHCP Snooping is not available. The reverse is also true; if DHCP Snooping is enabled, then DHCP Option-82 is not available. In addition, the following differences exist between these two features:

- DHCP Snooping does require and use the Option-82 data insertion capability, but does not implement any other behaviors defined in RFC 3046.
- DHCP Snooping is configurable at the switch level and on a per-VLAN basis, but DHCP Option-82 is only configurable at the switch level.

The following sections provide additional information about each DHCP security feature and how to configure feature parameters using the Command Line Interface (CLI).

Using the Relay Agent Information Option (Option-82)

This implementation of the DHCP relay agent information option (Option-82) feature is based on the functionality defined in RFC 3046. By default DHCP Option-82 functionality is disabled. The [ip helper agent-information](#) command is used to enable this feature at the switch level.

When this feature is enabled, communications between a DHCP client and a DHCP server are authenticated by the relay agent. To accomplish this task, the agent adds Option-82 data to the end of the options field in DHCP packets sent from a client to a DHCP server. Option-82 consists of two suboptions: Circuit ID and Remote ID. The agent fills in the following information for each of these suboptions:

- **Circuit ID**—the VLAN ID and slot/port from where the DHCP packet originated.
- **Remote ID**—the MAC address of the router interface associated with the VLAN ID specified in the Circuit ID suboption.

The DHCP Option-82 feature is only applicable when DHCP relay is used to forward DHCP packets between clients and servers associated with different VLANs. In addition, a secure IP network must exist between the relay agent and the DHCP server.

How the Relay Agent Processes DHCP Packets from the Client

The following table describes how the relay agent processes DHCP packets received from clients when the Option-82 feature is enabled for the switch:

If the DHCP packet from the client ...	The relay agent ...
Contains a zero gateway IP address (0.0.0.0) and no Option-82 data.	Inserts Option-82 with unique information to identify the client source.
Contains a zero gateway IP address (0.0.0.0) and Option-82 data.	<p>Drops the packet, keeps the Option-82 data and forwards the packet, or replaces the Option-82 data with its own Option-82 data and forwards the packet.</p> <p>The action performed by the relay agent in this case is determined by the agent information policy that is configured through the ip helper agent-information policy command.</p> <p>By default, this type of DHCP packet is dropped by the agent.</p>
Contains a non-zero gateway IP address and no Option-82 data.	Drops the packet without any further processing.
Contains a non-zero gateway IP address and Option-82 data.	Drops the packet if the gateway IP address matches a local subnet, otherwise the packet is forwarded without inserting Option-82 data.

How the Relay Agent Processes DHCP Packets from the Server

Note that if a DHCP server does not support Option-82, the server strips the option from the packet. If the server does support this option, the server will retain the Option-82 data received and send it back in a reply packet.

When the relay agent receives a DHCP packet from the DHCP server and the Option-82 feature is enabled, the agent will:

- 1** Extract the VLAN ID from the Circuit ID suboption field in the packet and compare the MAC address of the IP router interface for that VLAN to the MAC address contained in the Remote ID suboption field in the same packet.
- 2** If the IP router interface MAC address and the Remote ID MAC address are not the same, then the agent will drop the packet.
- 3** If the two MAC addresses match, then a check is made to see if the slot/port value in the Circuit ID suboption field in the packet matches a port that is associated with the VLAN also identified in the Circuit ID suboption field.
- 4** If the slot/port information does not identify an actual port associated with the Circuit ID VLAN, then the agent will drop the packet.
- 5** If the slot/port information does identify an actual port associated with the Circuit ID VLAN, then the agent strips the Option-82 data from the packet and unicasts the packet to the port identified in the Circuit ID suboption.

Enabling the Relay Agent Information Option-82

Use the **ip helper agent-information** command to enable the DHCP Option-82 feature for the switch. For example:

```
-> ip helper agent-information enable
```

This same command is also used to disable this feature. For example:

```
-> ip helper agent-information disable
```

Note that because this feature is not available on a per-VLAN basis, DHCP Option-82 functionality is not restricted to ports associated with a specific VLAN. Instead, DHCP traffic received on all ports is eligible for Option-82 data insertion when it is relayed by the agent.

Configuring a Relay Agent Information Option-82 Policy

As previously mentioned, when the relay agent receives a DHCP packet from a client that already contains Option-82 data, the packet is dropped by default. However, it is possible to configure a DHCP Option-82 policy that directs the relay agent to drop, keep, or replace the existing Option-82 data and then forward the packet to the server.

To configure a DHCP Option-82 policy, use the **ip helper agent-information policy** command. The following parameters are available with this command to specify the policy action:

- **drop**—The DHCP packet is dropped (the default).
- **keep**—The existing Option-82 data in the DHCP packet is retained and the packet is forwarded to the server.
- **replace**—The existing Option-82 data in the DHCP packet is replaced with local relay agent data and then forwarded to the server.

For example, the following commands configure DHCP Option-82 policies:

```
-> ip helper agent-information policy drop
```

```
-> ip helper agent-information policy keep
```

```
-> ip helper agent-information policy replace
```

Note that this type of policy applies to all DHCP packets received on all switch ports. In addition, if a packet that contains existing Option-82 data also contains a gateway IP address that matches a local subnet address, the relay agent will drop the packet and not apply any existing Option-82 policy.

Using DHCP Snooping

Using DHCP Snooping improves network security by filtering DHCP messages received from devices outside the network and building and maintaining a binding table (database) to track access information for such devices.

In order to identify DHCP traffic that originates from outside the network, DHCP Snooping categorizes ports as either trusted or untrusted. A port is trusted if it is connected to a device inside the network, such as a DHCP server. A port is untrusted if it is connected to a device outside the network, such as a customer switch or workstation.

Additional DHCP Snooping functionality provided includes the following:

- **Layer 2 DHCP Snooping**—Applies DHCP Snooping functionality to bridged DHCP client/server broadcasts without using the relay agent or requiring an IP interface on the client/server VLAN. See [“Layer 2 DHCP Snooping” on page 29-25](#) for more information.
- **IP Source Filtering**—Restricts DHCP Snooping port traffic to only packets that contain the client source MAC address and IP address. The DHCP Snooping binding table is used to verify the client information for the port that is enabled for IP source filtering. See [“Configuring Port IP Source Filtering” on page 29-23](#) for more information.
- **Rate Limiting**—Limits the rate of DHCP packets on the port. This functionality is achieved using the QoS application to configure ACLs for the port. See [Chapter 34, “Configuring QoS,”](#) in the *OmniSwitch 6250 Network Configuration Guide* for more information.

When DHCP Snooping is first enabled, all ports are considered untrusted. It is important to then configure ports connected to a DHCP server inside the network as trusted ports. See [“Configuring the Port Trust Mode” on page 29-22](#) for more information.

If a DHCP packet is received on an untrusted port, then it is considered an untrusted packet. If a DHCP packet is received on a trusted port, then it is considered a trusted packet. DHCP Snooping only filters untrusted packets and will drop such packets if one or more of the following conditions are true:

- The packet received is a DHCP server packet, such as a DHCPOFFER, DHCPACK, or DHCPNAK packet. When a server packet is received on an untrusted port, DHCP Snooping knows that it is not from a trusted server and discards the packet.
- The source MAC address of the packet and the DHCP client hardware address contained in the packet are not the same address.
- The packet is a DHCPRELEASE or DHCPDECLINE broadcast message that contains a source MAC address found in the DHCP Snooping binding table, but the interface information in the binding table does not match the interface on which the message was received.
- The packet includes a relay agent IP address that is a non-zero value.
- The packet already contains Option-82 data in the options field and the Option-82 check function is enabled. See [“Bypassing the Option-82 Check on Untrusted Ports” on page 29-22](#) for more information.

If none of the above are true, then DHCP Snooping accepts and forwards the packet. When a DHCPACK packet is received from a server, the following information is extracted from the packet to create an entry in the DHCP Snooping binding table:

- MAC address of the DHCP client.
- IP address for the client that was assigned by the DHCP server.

- The port from where the DHCP packet originated.
- The VLAN associated with the port from where the DHCP packet originated.
- The lease time for the assigned IP address.
- The binding entry type; dynamic or static (user-configured).

After extracting the above information and populating the binding table, the packet is then forwarded to the port from where the packet originated. Basically, the DHCP Snooping feature prevents the normal flooding of DHCP traffic. Instead, packets are delivered only to the appropriate client and server ports.

DHCP Snooping Configuration Guidelines

Consider the following when configuring the DHCP Snooping feature:

- Layer 3 DHCP Snooping requires the use of the relay agent to process DHCP packets. As a result, DHCP clients and servers must reside in different VLANs so that the relay agent is engaged to forward packets between the VLAN domains. See [“Configuring BOOTP/DHCP Relay Parameters” on page 29-10](#) for information about how to configure the relay agent on the switch.
- Layer 2 DHCP Snooping does not require the use of the relay agent to process DHCP packets. As a result, an IP interface is not needed for the client/server VLAN. See [“Layer 2 DHCP Snooping” on page 29-25](#) for more information.
- Both Layer 2 and Layer 3 DHCP Snooping are active when DHCP Snooping is globally enabled for the switch or enabled on one or more VLANs. See [“Enabling DHCP Snooping” on page 29-20](#) for more information.
- Configure ports connected to DHCP servers within the network as trusted ports. See [“Configuring the Port Trust Mode” on page 29-22](#) for more information.
- Make sure that Option-82 data insertion is always enabled at the switch or VLAN level. See [“Enabling DHCP Snooping” on page 29-20](#) for more information.
- DHCP packets received on untrusted ports that already contain the Option-82 data field are discarded by default. To accept such packets, configure DHCP Snooping to bypass the Option-82 check. See [“Bypassing the Option-82 Check on Untrusted Ports” on page 29-22](#) for more information.
- By default, rate limiting of DHCP traffic is done at a rate of 512 DHCP messages per second per switching ASIC. Each switching ASIC controls 24 ports (e.g. ports 1–24, 25–48, etc.) on a module.

Enabling DHCP Snooping

There are two levels of operation available for the DHCP Snooping feature: switch level or VLAN level. These two levels are exclusive of each other in that they both cannot operate on the switch at the same time. In addition, if the global DHCP relay agent information option (Option-82) is enabled for the switch, then DHCP Snooping at any level is not available. See [“Using the Relay Agent Information Option \(Option-82\)” on page 29-16](#) for more information.

Note. DHCP Snooping drops server packets received on untrusted ports (ports that connect to devices outside the network or firewall). It is important to configure ports connected to DHCP servers as trusted ports so that traffic to/from the server is not dropped.

Switch-level DHCP Snooping

By default, DHCP Snooping is disabled for the switch. To enable this feature at the switch level, use the **ip helper dhcp-snooping** command. For example:

```
-> ip helper dhcp-snooping enable
```

When DHCP Snooping is enabled at the switch level, all DHCP packets received on all switch ports are screened/filtered by DHCP Snooping. By default, only client DHCP traffic is allowed on the ports, unless the trust mode for a port is configured to block or allow all DHCP traffic. See [“Configuring the Port Trust Mode” on page 29-22](#) for more information.

In addition, the following functionality is also activated by default when switch-level DHCP Snooping is enabled:

- The DHCP Snooping binding table is created and maintained. To configure the status or add a static entry to this table, use the **ip helper dhcp-snooping binding** command.
- MAC address verification is performed to compare the source MAC address of the DHCP packet with the client hardware address contained in the packet. To configure the status of MAC address verification, use the **ip helper dhcp-snooping mac-address verification** command.
- Option-82 data is inserted into the packet and then DHCP reply packets are only sent to the port from where the DHCP request originated, instead of flooding these packets to all ports. To configure the status of Option-82 data insertion, use the **ip helper dhcp-snooping option-82 data-insertion** command.
- The base MAC address of the switch is inserted into the Circuit ID and Remote ID sub-options of the Option-82 field. To configure the type of data (base MAC address, system name, or user-defined) that is inserted into the Option-82 suboptions, use the **ip helper dhcp-snooping option-82 format** command.

Note the following when disabling DHCP Snooping functionality:

- Disabling Option-82 is not allowed if the binding table is enabled.
- Enabling the binding table is not allowed if Option-82 data insertion is not enabled at either the switch or VLAN level.

VLAN-Level DHCP Snooping

To enable DHCP Snooping at the VLAN level, use the **ip helper dhcp-snooping vlan** command. For example, the following command enables DHCP Snooping for VLAN 200:

```
-> ip helper dhcp-snooping vlan 200
```

When this feature is enabled at the VLAN level, DHCP Snooping functionality is only applied to ports that are associated with a VLAN that has this feature enabled. Up to 64 VLANs can have DHCP Snooping enabled. Note that enabling DHCP Snooping at the switch level is not allowed if it is enabled for one or more VLANs.

By default, when DHCP Snooping is enabled for a specific VLAN, MAC address verification and Option-82 data insertion is also enabled for the VLAN by default. To disable or enable either of these two features, use the **ip helper dhcp-snooping vlan** command with either the **mac-address verification** or **option-82 data-insertion** parameters. For example:

```
-> ip helper dhcp-snooping vlan 200 mac-address verification disable
```

```
-> ip helper dhcp-snooping vlan 200 option-82 data-insertion disable
```

Note that if the binding table functionality is enabled, disabling Option-82 data insertion for the VLAN is not allowed. See “[Configuring the DHCP Snooping Binding Table](#)” on page 29-23 for more information.

Note. If DHCP Snooping is *not* enabled for a VLAN, then all ports associated with the VLAN are considered trusted ports. VLAN-level DHCP Snooping does not filter DHCP traffic on ports associated with a VLAN that does not have this feature enabled.

Configuring the Port Trust Mode

The DHCP Snooping trust mode for a port determines whether or not the port accepts all DHCP traffic, client-only DHCP traffic, or blocks all DHCP traffic. The following trust modes for a port are configurable using the `ip helper dhcp-snooping port` command:

- **client-only**—The default mode applied to ports when DHCP Snooping is enabled. This mode restricts DHCP traffic on the port to only DHCP client-related traffic. When this mode is active for the port, the port is considered an untrusted interface.
- **trust**—This mode does not restrict DHCP traffic on the port. When this mode is active on a port, the port is considered a trusted interface. In this mode the port behaves as if DHCP Snooping is not enabled.
- **block**—This mode blocks all DHCP traffic on the port. When this mode is active for the port, the port is considered an untrusted interface.

To configure the trust mode for one or more ports, use the `ip helper dhcp-snooping port` command. For example, the following command changes the trust mode for port 1/12 to blocked:

```
-> ip helper dhcp-snooping port 1/12 block
```

It is also possible to specify a range of ports. For example, the following command changes the trust mode for ports 2/1 through 2/10 to trusted:

```
-> ip helper dhcp-snooping port 2/1-10 trust
```

Note that it is necessary to configure ports connected to DHCP servers within the network and/or firewall as trusted ports so that necessary DHCP traffic to/from the server is not blocked. Configuring the port mode as trusted also identifies the device connected to that port as a trusted device within the network.

Bypassing the Option-82 Check on Untrusted Ports

By default, DHCP Snooping checks packets received on untrusted ports (DHCP Snooping client-only or blocked ports) to see if the packets contain the Option-82 data field. If a packet does contain this field, the packet is dropped.

To allow untrusted ports to receive and process DHCP packets that already contain the Option-82 data field, use the `ip helper dhcp-snooping bypass option-82-check` command to disable the Option-82 check. For example:

```
-> ip helper dhcp-snooping bypass option-82-check enable
```

Configuring Port IP Source Filtering

IP source filtering applies to DHCP Snooping ports and restricts port traffic to only packets that contain the client source MAC address and IP address. The DHCP Snooping binding table is used to verify the client information for the port that is enabled for IP source filtering.

By default IP source filtering is disabled for a DHCP Snooping port. Use the [ip helper dhcp-snooping port ip-source-filtering](#) command to enable or disable this function for a specific port or range of ports. For example:

```
-> ip helper dhcp-snooping port 1/10 ip-source-filtering enable
-> ip helper dhcp-snooping port 2/1-5 ip-source-filtering enable
```

Note that when IP source filtering is enabled, the maximum number of clients supported is 125 per switching ASIC. Each switching ASIC controls 24 ports (e.g. ports 1–24, 25–48, etc.) on a module.

Configuring the DHCP Snooping Binding Table

The DHCP Snooping binding table is automatically enabled by default when DHCP Snooping is enabled at either the switch or VLAN level. This table is used by DHCP Snooping to filter DHCP traffic that is received on untrusted ports.

Entries are made in this table when the relay agent receives a DHCPACK packet from a trusted DHCP server. The agent extracts the client information, populates the binding table with the information and then forwards the DHCPACK packet to the port where the client request originated.

To enable or disable the DHCP Snooping binding table, use the [ip helper dhcp-snooping binding](#) command. For example:

```
-> ip helper dhcp-snooping binding enable
-> ip helper dhcp-snooping binding disable
```

Note that enabling the binding table functionality is not allowed if Option-82 data insertion is *not* enabled at either the switch or VLAN level.

In addition, it is also possible to configure static binding table entries. This type of entry is created using available [ip helper dhcp-snooping binding](#) command parameters to define the static entry. For example, the following command creates a static DHCP client entry:

```
-> ip helper dhcp-snooping binding 00:2a:95:51:6c:10 port 1/15 address
17.15.3.10 lease-time 3 vlan 200
```

To remove a static binding table entry, use the **no** form of the [ip helper dhcp-snooping binding](#) command. For example:

```
-> no ip helper dhcp-snooping binding 00:2a:95:51:6c:10 port 1/15 address
17.15.3.10 lease-time 3 vlan 200
```

To view the DHCP Snooping binding table contents, use the [show ip helper dhcp-snooping binding](#) command. See the *OmniSwitch CLI Reference Guide* for example outputs of this command.

Configuring the Binding Table Timeout

The contents of the DHCP Snooping binding table resides in the switch memory. In order to preserve table entries across switch reboots, the table contents is automatically saved to the **dhcpBinding.db** file located in the **/flash/switch** directory.

Note. Do not manually change the **dhcpBinding.db** file. This file is used by DHCP Snooping to preserve and maintain binding table entries. Changing the file name or contents can cause problems with this functionality or with the DHCP Snooping application itself.

The amount of time, in seconds, between each automatic save is referred to as the binding table timeout value. By default, the timeout value is 300 seconds. To configure this value, use the **ip helper dhcp-snooping binding timeout** command. For example, the following command sets the timeout value to 1500 seconds:

```
-> ip helper dhcp-snooping binding timeout 1500
```

Each time an automatic save is performed, the **dhcpBinding.db** file is time stamped.

Synchronizing the Binding Table

To synchronize the contents of the **dhcpBinding.db** file with the binding table contents that resides in memory, use the **ip helper dhcp-snooping binding action** command. This command provides two parameters: **purge** and **renew**. Use the **purge** parameter to clear binding table entries in memory and the **renew** parameter to populate the binding table with the contents of the **dhcpBinding.db** file. For example:

```
-> ip helper dhcp-snooping binding action purge
```

```
-> ip helper dhcp-snooping binding action renew
```

Synchronizing the binding table is only done when this command is used. There is no automatic triggering of this function. In addition, it is important to note that synchronizing the binding table loads **dhcpBinding.db** file contents into memory. This is the reverse of saving the binding table contents in memory to the **dhcpBinding.db** file, which is done at automatic time intervals as defined by the binding table timeout value. See “[Configuring the Binding Table Timeout](#)” on page 29-24 for more information.

Binding Table Retention

When the binding table is synchronized with the contents of the **dhcpBinding.db** file, any table entries with a MAC address that no longer appears in the MAC address table are cleared from the binding table. To retain these entries regardless of their MAC address table status, use the **ip helper dhcp-snooping binding persistency** command. For example:

```
-> ip helper dhcp-snooping binding persistency enable
```

When binding table retention is enabled, entries remain in the table for the term of their DHCP lease and are not removed even when the MAC address for the entry is cleared from the MAC address table.

To disable binding table retention, use the following command:

```
-> ip helper dhcp-snooping binding persistency disable
```

Use the **show ip helper** command to determine the status of binding table retention.

Layer 2 DHCP Snooping

By default, DHCP broadcasts are flooded on the default VLAN of the client/server port. If the DHCP client and server are both members of the same VLAN domain, the broadcast packets from these sources are bridged as Layer 2 traffic and not processed by the relay agent.

When DHCP Snooping is enabled at the switch level or for an individual VLAN, DHCP Snooping functionality is also applied to Layer 2 traffic. When DHCP Snooping is disabled at the switch level or disabled on the last VLAN to have snooping enabled on the switch, DHCP Snooping functionality is no longer applied to Layer 2 or Layer 3 traffic.

Verifying the DHCP Relay Configuration

To display information about the DHCP Relay and BOOTP/DHCP, use the **show** commands listed below.

For more information about the resulting displays from these commands, see the *OmniSwitch CLI Reference Guide*. An example of the output for the **show ip helper** command is also given in “[Quick Steps for Setting Up DHCP Relay](#)” on page 29-4.

show ip helper	Displays the current forward delay time, the maximum number of hops, the forwarding option, and each of the DHCP server IP addresses configured. Also displays the current configuration status for the DHCP relay agent information option (Option-82) and DHCP Snooping features.
show ip helper stats	Displays the number of packets the DHCP Relay service has received and transmitted, the number of packets dropped due to forward delay and maximum hops violations, and the number of packets processed since the last time these statistics were displayed.
show ip udp relay service	Displays the current configuration for UDP services by service name or by service port number.
show ip udp relay statistics	Displays the current statistics for each UDP port relay service. These statistics include the name of the service, the forwarding VLAN(s) configured for that service, and the number of packets the service has sent and received.
show ip udp relay destination	Displays the VLAN assignments to which the traffic received on the specified UDP service port is forwarded.
show ip helper dhcp-snooping vlan	Displays a list of VLANs that have DHCP Snooping enabled and whether or not MAC address verification and Option-82 data insertion is enabled for each VLAN.
show ip helper dhcp-snooping port	Displays the DHCP Snooping trust mode for the port and the number of packets destined for the port that were dropped due to a DHCP Snooping violation.
show ip helper dhcp-snooping binding	Displays the contents of the DHCP Snooping binding table (database).

30 Configuring Access Guardian

Access Guardian refers to the following Alcatel-Lucent security functions that work together to provide a dynamic, proactive network security solution:

- **Authentication and Classification**—Access control is configured on 802.1X-enabled ports using device classification policies. A policy can specify the use of one or more types of authentication methods (802.1X, MAC-based, or Web-based Captive Portal) for the same port. For each type of authentication, the policy also specifies the classification method (RADIUS, Group Mobility, default VLAN, or block device access).
- **User Network Profiles (UNP)**—A UNP defines network access for one or more user devices. Each device that is assigned to a specific profile is granted network access based on the profile criteria, instead of on an individual MAC address, IP address, or port.

In This Chapter

This chapter provides an overview of Access Guardian security features and describes how to configure these features through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

The following information and procedures are included in this chapter:

- [“Quick Steps for Configuring Access Guardian” on page 30-4](#)
- [“Access Guardian Overview” on page 30-7.](#)
- [“Interaction With Other Features” on page 30-11](#)
- [“Setting Up Port-Based Network Access Control” on page 30-12](#)
- [“Configuring Access Guardian Policies” on page 30-14](#)
- [“Configuring Captive Portal Authentication” on page 30-22](#)
- [“Configuring User Network Profiles” on page 30-29](#)

For more information about configuring 802.1X on switch ports, see [Chapter 32, “Configuring 802.1X.”](#)

Access Guardian Specifications

RFCs Supported	RFC 2284–PPP Extensible Authentication Protocol (EAP) RFC 2865–Remote Authentication Dial In User Service (RADIUS) RFC 2866–RADIUS Accounting RFC 2867–RADIUS Accounting Modifications for Tunnel Protocol Support RFC 2868–RADIUS Attributes for Tunnel Protocol Support RFC 2869–RADIUS Extensions
IEEE Standards Supported	IEEE 802.1X-2001–Standard for Port-based Network Access Control 802.1X RADIUS Usage Guidelines
Platforms Supported	OmniSwitch 6250

Access Guardian Defaults

The following default Access Guardian device classification policies are applied when 802.1x is enabled on a switch port:

Description	Keyword	Default Policy
Authentication and classification for 802.1x users (802.1x supplicants)	802.1x supplicant policy authentication	pass: group-mobility, default-vlan fail: block
Authentication and classification for non-802.1x users (non-supplicants).	802.1x non-supplicant policy authentication	block
Authentication and classification for web-based (Captive Portal) users.	802.1x captive-portal policy authentication	pass: default-vlan fail: block
Time limit for a Captive Portal session.	802.1x captive-portal session-limit	12 hours
Number of login attempts allowed per Captive Portal session.	802.1x captive-portal retry-count	3 login attempts
IP address for the Captive Portal login page	802.1x captive-portal address	10.123.0.1
Proxy web server URL for the Captive Portal user.	802.1x captive-portal proxy-server-url	proxy (Captive Portal looks for the word “proxy” to identify the web server URL.)

Quick Steps for Configuring Access Guardian

When 802.1x is enabled for a switch port, default Access Guardian device classification policies are applied to all devices connected to the port. As a result, it is only necessary to configure such policies if the default policy is not sufficient for network access control. Therefore, the following quick steps are optional but provide a brief tutorial for configuring Access Guardian policies:

- 1 To configure an Access Guardian policy that will authenticate and classify 802.1x users (supplicants), use the **802.1x supplicant policy authentication** command.

```
-> 802.1x 2/12 supplicant policy authentication pass group-mobility default-vlan
fail vlan 10 captive-portal
```

- 2 To configure an Access Guardian policy that will authenticate and classify non-802.1x users (non-supplicants), use the **802.1x non-supplicant policy authentication** command.

```
-> 802.1x 2/12 non-supplicant policy authentication pass group-mobility default-
vlan fail vlan 10 captive-portal
```

- 3 To configure an Access Guardian Captive Portal policy that will classify web-based clients, use the **802.1x captive-portal policy authentication** command. Note that this policy is triggered only when the Captive Portal option of a supplicant or non-supplicant policy is applied.

```
-> 802.1x 2/12 captive-portal policy authentication pass vlan 100 block fail
vlan 10
```

- 4 To configure the length of a Captive Portal session, use the **802.1x captive-portal session-limit** command.

```
-> 802.1x 3/1 captive-portal session-limit 8
```

- 5 To configure the number of Captive Portal login attempts allowed before a device is classified as a failed login, use the **802.1x captive-portal retry-count** command.

```
-> 802.1x 3/1 captive-portal retry-count 5
```

- 6 To bypass authentication and restrict device classification of non-802.1x users to VLANs that are not authenticated VLANs, use the **802.1x non-supplicant policy** command.

```
-> 802.1x 3/10 non-supplicant policy vlan 43 block
```

- 7 To set the Access Guardian policy back to the default classification policy for an 802.1x port, use the **802.1x policy default** command.

```
-> 802.1x 3/10 policy default
```

Note. Verify the Access Guardian configuration using the **show 802.1x device classification policies** command:

```
-> show 802.1x device classification policies

Device classification policies on 802.1x port 2/26
Supplicant:
  authentication:
    pass: group-mobility, default-vlan (default)
    fail: block (default)
Non-Supplicant:
  block (default)
```

```
Captive Portal:
  authentication:
    pass: default-vlan (default)
    fail: block (default)
Device classification policies on 802.1x port 2/48
Supplicant:
  authentication:
    pass: vlan 500, block
    fail: block (default)
Non-Supplicant:
  block (default)
Captive Portal:
  authentication:
    pass: default-vlan (default)
    fail: block (default)
```

To verify the Captive Portal configuration for an 802.1X-enabled port, use the [802.1x auth-server-down](#) command:

```
-> show 802.1x 1/13

802.1x configuration for slot 1 port 13:

direction                               = both,
operational directions                   = both,
port-control                             = auto,
quiet-period (seconds)                   = 60,
tx-period (seconds)                       = 30,
supp-timeout (seconds)                    = 30,
server-timeout (seconds)                  = 30,
max-req                                   = 2,
re-authperiod (seconds)                   = 3600,
reauthentication                         = no
Supplicant polling retry count            = 2
Captive Portal Session Limit (hrs)        = 12
Captive Portal Login Retry Count          = 3
```

To verify the global Captive Portal configuration for the switch, use the [show 802.1x auth-server-down](#) command:

```
-> show 802.1x captive-portal configuration

802.1x Captive Portal configuration for slot 7 port 11:

Session Limit (hours)                    = 4,
Login Retry Count                         = 5,

802.1x Captive Portal configuration for slot 8 port 1:

Session Limit (hours)                    = 8,
Login Retry Count                         = 2,
```

To display the number of non-802.1x users learned on the switch, use the **show 802.1x non-supplicant** command:

```
-> show 802.1x non-supplicant
```

Slot Port	MAC Address	Authentication Status	Classification Policy	Vlan Learned
03/3	00:61:22:15:22:33	Failed	Vlan ID	1001
03/3	00:61:22:44:75:66	Authenticated	MAC Authent	14
03/11	00:00:39:47:4f:0c	Failed	Vlan ID	1001
03/11	00:00:39:c9:5a:0c	Authenticated	Group Mobility	12
03/11	00:b0:d0:52:47:35	Authenticated	Group Mobility	12
03/11	00:c0:4f:0e:70:68	Authenticated	MAC Authent	14

See the *OmniSwitch 6250 CLI Reference Guide* for information about the fields in this display.

Quick Steps for Configuring User Network Profiles

A User Network Profile (UNP) is a configurable option for Access Guardian device classification policies. The following quick steps provide a brief tutorial on how to create a UNP and configure a device classification policy to use the UNP to classify a device:

- 1 To create a User Network Profile, use the **aaa user-network-profile** command.

```
-> aaa user-network-profile name guest_user vlan 500
```

Note. Verify the UNP configuration using the **show aaa user-network-profile** command:

```
-> show aaa user-network-profile
```

Role Name	Vlan
guest-user	500
accounting	20

See the *OmniSwitch 6250 CLI Reference Guide* for information about the fields in this display.

Access Guardian Overview

Access Guardian is a combination of authentication, device compliance, and access control functions that provide a *proactive* solution to network security. Implemented through the switch hardware and software, Access Guardian helps administrators:

- Determine who is on the network.
- Check if end users are compliant.
- Direct what end users can access within the network.

The following switch-based features provide the Access Guardian functionality:

- 802.1X, MAC, and Captive Portal authentication.
- 802.1X device classification policies.

Authentication and Classification

Physical devices attached to a LAN port on the switch through a point-to-point LAN connection may be authenticated through the switch using port-based network access control. This control is available through the IEEE 802.1X standard implemented on the switch.

Access Guardian uses this implementation of 802.1X to provide configurable device classification policies for authenticating both 802.1x clients (supplicants) and non-802.1x clients (non-supplicants). Such policies include the following options for authentication:

- **802.1X authentication for supplicants.**

Uses Extensible Authentication Protocol (EAP) between end device and network device (NAS) to authenticate the supplicant via a RADIUS server. If authentication returns a VLAN ID, the supplicant is assigned to that VLAN. If a VLAN ID is not returned or authentication fails, then the device classification policy configuration for the port provides the network access control for the supplicant.

- **MAC-based authentication for non-supplicants.**

MAC-based authentication requires no agent or special protocol on the non-suppliant device; the source MAC address of the device is verified via a remote RADIUS server. The switch sends RADIUS frames to the server with the source MAC address embedded in the username and password attributes. If authentication returns a VLAN ID, the non-suppliant is assigned to that VLAN. If a VLAN ID is not returned or authentication fails, then the device classification policy configuration for the port provides the network access control for the non-suppliant.

- **Captive Portal Web-based authentication for supplicants and non-supplicants.**

Captive Portal is a configurable option for both supplicant and non-suppliant policies. When the Captive Portal option is invoked, a Web page is presented to the user device to prompt the user to enter login credentials. If authentication returns a VLAN ID, the device is assigned to that VLAN. If a VLAN ID is not returned or authentication fails, a separate Captive Portal policy then determines the network access control for the supplicant or non-suppliant.

The authentication functionality provided through device classification policies allows the administrator to dynamically assign the appropriate method of authentication regardless of how many users are connected to a port or the type of user (for example, IP phones). In other words, multiple authentication methods for multiple users are supported on the same port.

Device classification policies are applied to each device connected to an 802.1X port until the appropriate method of authentication is determined. For example:

- An 802.1X capable device is challenged to provide credentials required for 802.1X authentication.
- A non-802.1X device, such as a printer, is not challenged but identified using MAC-based authentication.
- A device that fails authentication is prompted to provide credentials using Captive Portal.

Using Device Classification Policies

In addition to authentication, Access Guardian device classification policies are used to determine which of the following actions are applied to a device if authentication does not return a VLAN ID, authentication fails, or no authentication is performed:

- Assign the user device to a specific VLAN. For example, all guest users are assigned to VLAN 500 or are only allowed access to the default VLAN of the 802.1X port to which the device is connected.
- Use Group Mobility to dynamically assign a device to a VLAN. VLAN rules are used by Group Mobility to classify user devices.
- Do not perform any type of authentication on the device; only apply classification policies to determine what the end user can access on the network.
- Redirect the end user device to a Web-based login page for authentication.
- Block the device from accessing the network.

Device Classification Policy Types

There are four types of Access Guardian device classification policies: 802.1X authentication (supplicants), MAC-based authentication (non-supplicants), Captive Portal authentication (supplicant and non-supplicant), and non-supplicant (no authentication). These policies provide the following configurable policy options for classifying devices:

- 1 Captive Portal**—redirects the user device to a Web-based login screen and requires the user to enter credentials to gain network access. This option is used only with the 802.1X, MAC, or Non-supplicant policies. The Captive Portal policy is applied after Web-based authentication is attempted, so this option is not valid for Captive Portal policies. See [“Configuring the Captive Portal Policy” on page 30-20](#).
- 2 Group Mobility**—uses Group Mobility VLAN rules to determine the VLAN assignment for a device.
- 3 VLAN ID**—assigns the device to the specified VLAN.
- 4 Default VLAN**—assigns a device to the default VLAN for the 802.1x port.
- 5 Block**—blocks a device from accessing the 802.1x port.

It is possible to configure one or more of the above options for a single policy. The order in which the policy options are applied to a device is determined by the order in which the option was configured. For example, if a MAC-based authentication policy is configured to use the Group Mobility and default VLAN options, then the policy actions are applied in the following sequence:

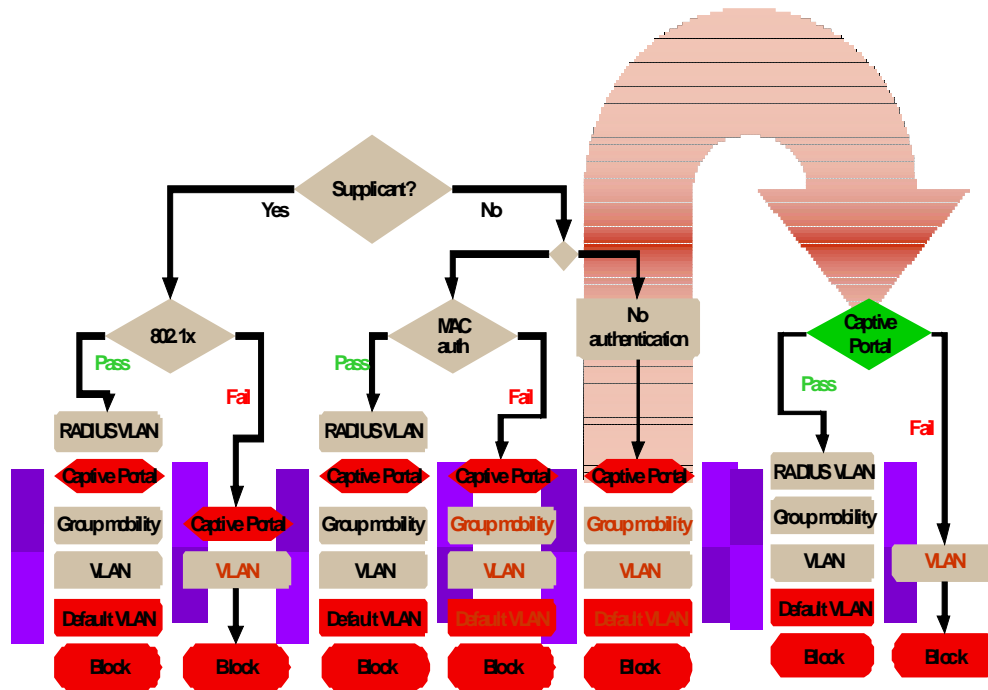
- 1** MAC-based authentication is performed.
- 2** If authentication was successful and provided a VLAN ID, the client is assigned to that VLAN and no further policy options are applied.

- 3 If a VLAN ID was not provided or authentication failed, then Group Mobility applies VLAN rules.
- 4 If there are no Group Mobility VLAN rules that match the client traffic, then the device is learned in the default VLAN for the 802.1X port.

See [“Configuring Access Guardian Policies” on page 30-14](#) for more information about how to use and configure policies.

Note. It is possible to bypass 802.1x authentication and classify supplicants connected to an 802.1x port as non-supplicants (see the [“Configuring the Number of Polling Retries”](#) section in [Chapter 32, “Configuring 802.1X,”](#) for more information). When this is done, all devices (including supplicants) are then classified as non-supplicants. As a result, non-supplicant policies that use MAC-based authentication are now applicable to supplicant devices, not just non-supplicant devices.

The following diagram illustrates the conceptual flow of Access Guardian policies, including the separate Web-based authentication branch provided by Captive Portal:



Access Guardian Policy Flow

As shown in the above diagram, Captive Portal is an optional policy that is available for both supplicant and non-supplicant policies. When successful RADIUS authentication does not return a VLAN ID or a device fails authentication, policies configured for the port are examined. If the Captive Portal policy is configured for the port and invoked by device traffic, then the user must authenticate through the switch via standard web browser software.

For more information, see [“Configuring Access Guardian Policies” on page 30-14](#) and [“Configuring Captive Portal Authentication” on page 30-22](#).

User Network Profiles (Role-Based Access)

A User Network Profile (UNP) defines network access for one or more user devices. Each device that is assigned to a specific profile is granted network access based on the profile criteria, instead of on an individual MAC address, IP address, or port.

Assigning users to a profile provides greater flexibility and scalability across the network. Administrators can use profiles to group users according to function. All users assigned to the same UNP become members of that profile group. The UNP then determines what network access resources are available to a group of users, regardless of source subnet, VLAN or other characteristics.

A User Network Profile consists of the following attributes:

- **UNP name.** The UNP name is obtained from the RADIUS server and mapped to the same profile name configured on the switch. The switch profile then identifies the VLAN ID attribute value.
- **VLAN ID.** All members of the profile group are assigned to the VLAN ID specified by the profile.

An administrator can implement the same UNP name across the entire network infrastructure, as the VLAN association is kept locally on each switch. For example, the administrator can deploy the UNP named “Engineering” in one building using VLAN 10, while the same UNP deployed in another building can use VLAN 20. The same UNP access is applied to all profile users in each building, even though they belong to different VLANs.

User profiles must already exist in the switch configuration before they are deployed via 802.1x or MAC-based authentication. See [“Configuring User Network Profiles” on page 30-29](#).

Interaction With Other Features

This section contains important information about how other OmniSwitch features interact with Access Guardian. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

Captive Portal - Browser Support

The Captive Portal authentication feature presents the user with a Web page for entering login credentials. The following browsers are supported for Captive Portal users:

- Internet Explorer 6 or later
- Firefox2 or later

Setting Up Port-Based Network Access Control

For port-based network access control, 802.1X must be enabled for the switch and the switch must know which servers to use for authenticating 802.1X supplicants and non-supplicants.

In addition, 802.1X must be enabled on each port that is connected to a n 802.1X supplicant (or device). Optional parameters may be set for each 802.1X port.

The following sections describe these procedures in detail.

Setting 802.1X Switch Parameters

Use the **aaa authentication 802.1x** command to enable 802.1X for the switch and specify an authentication server (or servers) to be used for authenticating 802.1X ports. The servers must already be configured through the **aaa radius-server** command. An example of specifying authentication servers for authenticating all 802.1X ports on the switch:

```
-> aaa authentication 802.1x rad1 rad2
```

In this example, the **rad1** server will be used for authenticating 802.1X ports. If **rad1** becomes unavailable, the switch will use **rad2** for 802.1X authentication. When this command is used, 802.1X is automatically enabled for the switch.

Enabling MAC Authentication

Use the **aaa authentication mac** command to enable MAC authentication for the switch and specify an authentication server (or servers) to be used for authenticating non-supplicants on 802.1x ports. As with enabling 802.1x authentication, the servers specified with this command must already be configured through the **aaa radius-server** command.

The following example command specifies authentication servers for authenticating non-supplicant devices on 802.1x ports:

```
-> aaa authentication mac rad1 rad2
```

Note that the same RADIUS servers can be used for 802.1x (supplicant) and MAC (non-supplicant) authentication. Using different servers for each type of authentication is allowed but not required.

For more information about using MAC authentication and classifying non-supplicant devices, see [“Authentication and Classification” on page 30-7](#), [“Configuring Access Guardian Policies” on page 30-14](#), and [“Configuring User Network Profiles” on page 30-29](#).

Enabling an Authentication Server Down Policy

An authentication server down policy is used to classify devices attempting to authenticate through 802.1.x switch ports when the RADIUS server is unreachable. This type of policy offers two options:

- Assign the device to a pre-configured User Network Profile (UNP). See [“Configuring User Network Profiles” on page 30-29](#) for more information.
- Block access to the switch; device traffic is dropped.

A default authentication server down policy is configured to block device access. To change the policy configuration, use the **802.1x auth-server-down** command. For example:

```
-> 802.1x auth-server-down policy user-network-profile tem_unp1
```

The **802.1x auth-server-down** command is also used to enable or disable a policy. For example:

```
-> 802.1x auth-server-down enable
-> 802.1x auth-server-down disable
```

After a device is classified according to an authentication server down policy, an attempt to re-authenticate the device is made after a specific period of time (30 seconds by default). This time value is configurable using the **802.1x auth-server-down re-authperiod** command. For example:

```
-> 802.1x auth-server-down re-authperiod 500
```

The authentication server down policy and re-authentication time period configuration applies to all 802.1x ports on the switch. To verify the authentication server down policy configuration, use the **show 802.1x auth-server-down** command.

Note that when device authentication fails due to an unreachable RADIUS server, an event message is sent to the switch logging utility (swlog). See [Chapter 39, “Using Switch Logging,”](#) for more information.

Enabling 802.1X on Ports

To enable 802.1X on a port, use the **vlan port 802.1x** command. The port must first be configured as a mobile port.

```
-> vlan port mobile 3/1
-> vlan port 3/1 802.1x enable
```

The **vlan port 802.1x** command enables 802.1X on port 1 of slot 3. The port will be set up with defaults listed in “802.1X Defaults” of the [Chapter 32, “Configuring 802.1X.”](#)

To disable 802.1X on a port, use the **disable** option with **vlan port 802.1x** command. For more information about **vlan port** commands, See [Chapter 6, “Assigning Ports to VLANs.”](#)

Configuring 802.1X Port Parameters

By default, when 802.1X is enabled on a port, the port is configured for bidirectional control, automatic authorization, and re-authentication. In addition, there are several timeout values that are set by default as well as a maximum number of times the switch will retransmit an authentication request to the user.

If it is necessary to change the default values of these parameters, see [Chapter 32, “Configuring 802.1X,”](#) for information about how to configure 802.1X port parameters.

Configuring Access Guardian Policies

The Access Guardian provides functionality that allows the configuration of 802.1x device classification policies for supplicants (802.1x clients) and non-supplicants (non-802.1x clients). See [“Device Classification Policy Types” on page 30-8](#) for more information.

Configuring device classification policies is only supported on mobile, 802.1x-enabled ports. In addition, the port control status for the port must allow auto authorization (the default). See the [“Configuring the Port Authorization”](#) section in [Chapter 32, “Configuring 802.1X,”](#) for specific information about how to enable 802.1x functionality on a port.

As described in [“Device Classification Policy Types” on page 30-8](#), there are several types of policy options that when combined together create either a supplicant or non-supplicant policy. Consider the following when configuring policies:

- A single policy option can only appear once for a pass condition and once for a failed condition in a single policy.
- Up to three VLAN ID policy options are allowed within the same policy, as long as the ID number is different for each instance specified (e.g., VLAN 20 VLAN 30 VLAN 40).
- A policy must terminate. The last policy option must result in either blocking the device, assigning the device to the default VLAN, or invoking Captive Portal for web-based authentication. If a final policy option is not specified, the block option is used by default.
- The order in which policy options are configured determines the order in which they are applied to the device.

The following table provides examples of policies that were incorrectly configured and a description of the problem:

Incorrect Policy Command	Problem
802.1x 1/45 supplicant policy authentication pass group-mobility vlan 200 group-mobility fail block	The group-mobility option is specified more than once as a pass condition.
802.1x 1/24 non-supplicant policy authentication pass vlan 20 vlan 30 vlan 40 vlan 50 fail block	More than three VLAN ID options are specified in the same command.

Note that if no policies are configured on an 802.1x port, access from non-supplicant devices is blocked and the following default classification policy is applied to supplicant devices:

- 1 802.1x authentication via remote RADIUS server is attempted.
- 2 If authentication fails or successful authentication returns a VLAN ID that does not exist, the device is blocked.
- 3 If authentication is successful and returns a VLAN ID that exists in the switch configuration, the supplicant is assigned to that VLAN.
- 4 If authentication is successful but does not return a VLAN ID, Group Mobility checks if there are any VLAN rules or User Network Profile mobile rules that will classify the supplicant.
- 5 If Group Mobility classification fails, the supplicant is assigned to the default VLAN ID for the 802.1x port.

Configuring Supplicant Policies

Supplicant policies are used to classify 802.1x devices connected to 802.1x-enabled switch ports when 802.1x authentication does not return a VLAN ID or authentication fails. To configure supplicant policies, use the **802.1x supplicant policy authentication** command. The following parameter keywords are available with this command to specify policy options for classifying devices:

supplicant policy keywords

group mobility
vlan
default-vlan
block
captive-portal
pass
fail

If no policy keywords are specified with this command (for example, **802.1x 1/10 supplicant policy authentication**), then supplicants are blocked if 802.1x authentication fails or does not return a VLAN ID.

Note that the order in which parameters are configured determines the order in which they are applied. For example, the following commands apply Group Mobility rules at different times during the classification process:

```
-> 802.1x 2/12 supplicant policy authentication pass group-mobility vlan 10
block fail vlan 10 default-vlan

-> 802.1x 2/12 supplicant policy authentication pass vlan 10 group-mobility
block fail vlan 10 default-vlan
```

The first command in the above example checks Group Mobility rules first then checks for VLAN 10 next. The second command checks for VLAN 10 first then checks for Group Mobility rules.

Use the **pass** keyword to specify which options to apply when 802.1x authentication is successful but does not return a VLAN ID. Use the **fail** keyword to specify which options to apply when 802.1x authentication fails or returns a VLAN ID that does not exist. The **pass** keyword is implied and therefore an optional keyword. If the **fail** keyword is not used, the default action is to block the device.

Note. When a policy option is configured as a fail condition, device classification is restricted to assigning supplicant devices to VLANs that are *not* authenticated VLANs.

Supplicant Policy Examples

The following table provides example supplicant policy commands and a description of how the resulting policy is applied to classify supplicant devices:

Supplicant Policy Command Example	Description
802.1x 1/24 supplicant policy authentication pass group-mobility default-vlan fail vlan 43 block	<p>If the 802.1x authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility rules are applied. 2 If Group Mobility classification fails, then the device is assigned to the default VLAN for port 1/24. <p>If the device fails 802.1x authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 If VLAN 43 exists and is not an authenticated VLAN, then the device is assigned to VLAN 43. 2 If VLAN 43 does not exist or is an authenticated VLAN, then the device is blocked from accessing the switch on port 1/24.
802.1x 1/48 supplicant policy authentication group-mobility vlan 127 default-vlan	<p>If the 802.1x authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility rules are applied. 2 If Group Mobility classification fails, then the device is assigned to VLAN 127. 3 If VLAN 127 does not exist, then the device is assigned to the default VLAN for port 1/48. <p>If the device fails 802.1x authentication, the device is blocked on port 1/48.</p>
802.1x 2/12 supplicant policy authentication pass group-mobility captive-portal fail vlan 10 captive-portal	<p>If the 802.1x authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility rules are applied. 2 If Group Mobility classification fails, then the user is prompted to enter a user name and password through a web-based portal. <p>If the device fails 802.1x authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 If VLAN 10 exists and is not an authenticated VLAN, then the device is assigned to VLAN 10. 2 If VLAN 10 does not exist or is an authenticated VLAN, then the user is prompted to enter a user name and password through a web-based portal.

Supplicant Policy Command Example	Description
802.1x 2/1 supplicant policy authentication fail captive-portal	<p>If the 802.1x authentication process is successful but does not return a VLAN ID, the user is blocked from accessing the switch on port 2/1.</p> <p>If the device fails 802.1x authentication, then the user is prompted to enter a user name and password through a web-based portal.</p>

Configuring Non-supplicant Policies

Non-supplicant policies are used to classify non-802.1x devices connected to 802.1x-enabled switch ports. There are two types of non-supplicant policies. One type uses MAC authentication to verify the non-802.1x device. The second type does not perform any authentication and limits device assignment only to those VLANs that are not authenticated VLANs.

To configure a non-supplicant policy that will perform MAC authentication, use the **802.1x non-supplicant policy authentication** command. The following parameter keywords are available with this command to specify one or more policy options for classifying devices:

supplicant policy keywords

group-mobility
vlan
default-vlan
block
captive-portal
pass
fail

The order in which parameters are configured determines the order in which they are applied. For example, the following commands apply Group Mobility rules at different times during the classification process:

```
-> 802.1x 2/12 non-supplicant policy authentication pass group-mobility vlan 10
block fail vlan 10 default-vlan

-> 802.1x 2/12 non-supplicant policy authentication pass vlan 10 group-mobility
block fail vlan 10 default-vlan
```

The first command in the above example checks Group Mobility rules first then checks for VLAN 10 next. The second command checks for VLAN 10 first then checks for Group Mobility rules.

Use the **pass** keyword to specify which options to apply when 802.1x authentication is successful but does not return a VLAN ID. Use the **fail** keyword to specify which options to apply when 802.1x authentication fails or returns a VLAN ID that does not exist. The **pass** keyword is implied and therefore an optional keyword. If the **fail** keyword is not used, the default action is to block the device.

Use the **pass** keyword to specify which options to apply when MAC authentication is successful but does not return a VLAN ID. Use the **fail** keyword to specify which options to apply when MAC authentication fails. The **pass** keyword is implied and therefore an optional keyword. If the **fail** keyword is not used, the default action is to block the device when authentication fails.

Note. When a policy option is configured as a fail condition, device classification is restricted to assigning supplicant devices to VLANs that are *not* authenticated VLANs.

To configure a non-suppliant policy that will *not* perform MAC authentication, use the **802.1x non-suppliant policy** command. The following parameter keywords are available with this command to specify one or more policies for classifying devices:

suppliant policy keywords

group-mobility
 vlan
 default-vlan
 block
 captive-portal

Note that this type of policy does not use 802.1x or MAC authentication. As a result, all of the available policy keywords restrict the assignment of the non-suppliant device to only those VLANs that are *not* authenticated VLANs. The **pass** and **fail** keywords are not used when configuring this type of policy.

Non-suppliant Policy Examples

The following table provides example non-suppliant policy commands and a description of how the resulting policy is applied to classify supplicant devices:

Suppliant Policy Command Example	Description
802.1x 1/24 non-suppliant policy authentication pass group-mobility default-vlan fail vlan 10 block	<p>If the MAC authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility VLAN or UNP mobile rules are applied. 2 If Group Mobility classification fails, then the device is assigned to the default VLAN for port 1/24. <p>If the device fails MAC authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 If VLAN 10 exists and is not an authenticated VLAN, the device is assigned to VLAN 10. 2 If VLAN 10 does not exist or is an authenticated VLAN, the device is blocked from accessing the switch on port 1/24.
802.1x 1/48 non-suppliant policy authentication vlan 10 default-vlan	<p>If the MAC authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 The device is assigned to VLAN 10. 2 If VLAN 10 does not exist, then the device is assigned to the default VLAN for port 1/48. <p>If the device fails MAC authentication, the device is blocked from accessing the switch on port 1/48.</p>

Supplicant Policy Command Example	Description
802.1x 2/1 non-supplicant policy authentication fail vlan 100 default-vlan	<p>If MAC authentication does not return a VLAN ID, the device is blocked from accessing the switch on port 2/1.</p> <p>If the device fails MAC authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 If VLAN 100 exists and is not an authenticated VLAN, the device is assigned to VLAN 100. 2 If VLAN 100 does not exist or is an authenticated VLAN, the device is assigned to the default VLAN for port 2/1. 3 If the default VLAN for port 2/1 is an authenticated VLAN, then the device is blocked from accessing the switch on port 2/1.
802.1x 2/10 non-supplicant policy authentication pass vlan 10 block fail group-mobility default-vlan	<p>If the MAC authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 The device is assigned to VLAN 10. 2 If VLAN 10 does not exist, then the device is blocked from accessing the switch on port 2/10. <p>If the device fails MAC authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility VLAN or UNP mobile rules are applied. 2 If Group Mobility classification fails, then the device is assigned to the default VLAN for port 2/10. 3 If the default VLAN for port 2/10 is an authenticated VLAN, then the device is blocked from accessing the switch on port 2/10.
802.1x 3/1 non-supplicant policy authentication pass vlan 10 block fail group-mobility vlan 43 default-vlan	<p>If the MAC authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 The device is assigned to VLAN 10. 2 If VLAN 10 does not exist, then the device is blocked from accessing the switch on port 3/1. <p>If the device fails MAC authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility VLAN or UNP mobile rules are applied. 2 If Group Mobility classification fails, then the device is assigned to VLAN 43. 3 If VLAN 43 does not exist or is an authenticated VLAN, then the device is assigned to the default VLAN for port 3/1. 4 If the default VLAN for port 3/1 is an authenticated VLAN, then the device is blocked from accessing the switch on port 3/1.

Supplicant Policy Command Example	Description
802.1x 2/12 non-supplicant policy authentication pass group-mobility captive-portal fail vlan 10 captive-portal	<p>If the MAC authentication process is successful but does not return a VLAN ID for the device, then the following occurs:</p> <ol style="list-style-type: none"> 1 Group Mobility VLAN or UNP mobile rules are applied. 2 If Group Mobility classification fails, then the user is prompted to enter a user name and password through a web-based portal. <p>If the device fails MAC authentication, then the following occurs:</p> <ol style="list-style-type: none"> 1 If VLAN 10 exists and is not an authenticated VLAN, then the device is assigned to VLAN 10. 2 If VLAN 10 does not exist or is an authenticated VLAN, then the user is prompted to enter a user name and password through a web-based portal.
802.1x 3/1 non-supplicant policy authentication fail captive-portal	<p>If MAC authentication does not return a VLAN ID, the device is blocked from accessing the switch on port 3/1.</p> <p>If the device fails 802.1x authentication, then the user is prompted to enter a user name and password through a web-based portal.</p>
802.1x 3/10 non-supplicant policy vlan 43 block	<p>No authentication process is performed, but the following classification still occurs:</p> <ol style="list-style-type: none"> 1 If VLAN 43 exists and is not an authenticated VLAN, then the device is assigned to VLAN 43. 2 If VLAN 43 does not exist or is an authenticated VLAN, then the device is blocked from accessing the switch on port 3/10.

Configuring the Captive Portal Policy

The Captive Portal device classification policy is similar to supplicant and non-supplicant policies in that it determines the VLAN assignment for devices that were not assigned a VLAN through authentication or for devices that failed 802.1x or MAC authentication. The difference is that the Captive Portal policy is only invoked as a result of web-based authentication; supplicant and non-supplicant policies are triggered off of 802.1x port-based authentication.

Web-based authentication is configured by specifying Captive Portal as a pass or fail case for port-based supplicant and non-supplicant policies (see [“Configuring Supplicant Policies” on page 30-15](#) and [“Configuring Non-supplicant Policies” on page 30-17](#) for more information). When the web-based authentication process is complete, the Captive Portal policy classifies the device into a specific VLAN based on the results of that process.

When 802.1x is enabled for a port, a default supplicant, non-supplicant, and Captive Portal policy is automatically configured for the port. The default Captive Portal policy assigns a device to the default VLAN for the port if authentication was successful but did not return a VLAN ID or blocks a device on the port if

the device failed authentication. As a result, it is only necessary to change the policy if the default pass and fail cases are not sufficient.

To change the Captive Portal policy configuration, use the **802.1x captive-portal policy authentication** command. The following keywords are available with this command to specify one or more policies for classifying devices.

Captive Portal keywords

group-mobility
vlan
default-vlan
block
captive-portal
pass
fail

Note the following when configuring Captive Portal policies:

- The **captive-portal** parameter is not an option with this type of policy, as it is not possible to next Captive Portal policies. In addition, the **captive-portal** parameter is used only in supplicant and non-supplicant policies to invoke web-based authentication, not to classify a device for VLAN assignment.
- The order in which parameters are configured determines the order in which they are applied. For example, the following commands apply Group Mobility rules at different times during the classification process:

```
-> 802.1x 2/12 captive-portal policy authentication pass group-mobility vlan 10  
block fail vlan 10 default-vlan
```

```
-> 802.1x 2/12 captive-portal policy authentication pass vlan 10 group-mobility  
block fail vlan 10 default-vlan
```

The first command in the above example checks Group Mobility rules first then checks for VLAN 10 next. The second command checks for VLAN 10 first then checks for Group Mobility rules.

- When a policy is specified as a policy to apply when authentication fails, device classification is restricted to assigning non-supplicant devices to VLANs that are *not* authenticated VLANs

Configuring Captive Portal Authentication

Captive Portal authentication allows Web browser clients to authenticate through the switch using 802.1x or MAC authentication via a RADIUS server. The following configuration tasks describe how to set up Captive Portal authentication for the switch and on client devices:

- **Avoid using the 10.123.0.0/16 subnet within the network.** This subnet is used exclusively by the Captive Portal feature to redirect DNS requests to the Captive Portal login screen (Captive Portal IP 10.123.0.1) and to assign a temporary IP address for a client device that is attempting web-based authentication.

If a different Captive Portal subnet is required to avoid a conflict within the IP network, use the [802.1x captive-portal address](#) command to change the second octet of this IP address. Note that the second octet is the only configurable part of the Captive Portal IP address that is allowed.

- **Make sure a standard browser is available on the client device.** No specialized client software is required. The following Web browser software is supported (note that only HTTPS is supported at this time):

Platform	Web Browser Software	Java Version
Windows XP	IE6 and IE7; Firefox2 and Firefox3	Java 1.6 updates 5 through 12
Windows Vista	IE7; Firefox2 and Firefox3	Java 1.6 updates 5 through 12
Linux	Firefox2 and Firefox3	Java 1.6 updates 5 through 12

- **Configure the homepage URL for the client browser.** The Captive Portal authentication process responds only to browser queries that contain the “**www**”, “**http**”, or “**https**” prefix in the URL. As a result, it is necessary to configure the homepage URL for the browser with at least one of these three prefixes.
- **Configure a specific proxy server URL.** Captive Portal looks for the word “proxy” to identify the proxy server URL used by the client. If this URL does not contain the word “proxy”, use the [802.1x captive-portal proxy-server-url](#) command to specify the URL address to use.
- **Configure an 802.1x device classification policy for Captive Portal authentication.** A supplicant or non-supplicant policy configured with Captive Portal as a pass or fail condition is required to invoke Captive Portal authentication. For more information, see “[Configuring Supplicant Policies](#)” on page 30-15 and “[Configuring Non-supplicant Policies](#)” on page 30-17.
- **Configure a Captive Portal device classification policy.** A separate Captive Portal policy is required to classify devices when successful web-based authentication does not return a VLAN ID or authentication fails. For more information, see “[Configuring the Captive Portal Policy](#)” on page 30-20.
- **Configure the Captive Portal session time limit.** This time limit determines the length of the Captive Portal login session. When this time limit expires, the user is automatically logged out and network access is blocked. For more information, see “[Configuring Captive Portal Session Parameters](#)” on page 30-23.
- **Configure the number of Captive Portal login attempts allowed.** This number determines the number of failed login attempts a user is allowed when initiating a Captive Portal session. For more information, see “[Configuring Captive Portal Session Parameters](#)” on page 30-23.

Configuring Captive Portal Session Parameters

When 802.1x is enabled for the port, the default session time limit and retry count values are automatically applied to any Captive Portal session initiated on the port. As a result, it is only necessary to configure these parameters if the default values are not sufficient.

The **802.1x captive-portal session-limit** command is used to configure the amount of time a Captive Portal session remains active after a successful login. At the end of this time, the user is automatically logged out of the session and no longer has network access. By default, the session limit is set to 12 hours. To allow a user to remain logged in for an indefinite amount of time, specify 0 for this parameter value.

```
-> 802.1x 1/10 captive-portal session-limit 0
```

The **802.1x captive-portal retry-count** command is used to configure the maximum number of times a user can try to login through the Captive Portal login web page. When this limit is reached without achieving a successful login, the fail case of the Captive Portal device classification policy configured for the 802.1x port is applied to the user device. The default login retry count is set to 3. To specify an unlimited amount of login retries, specify 0 for this parameter value.

```
-> 802.1x 1/10 captive-portal retry-count 0
```

Use the **802.1x auth-server-down** command to display the current values for the Captive Portal session parameters. An example of this command is available in the [“Quick Steps for Configuring Access Guardian”](#) on page 30-4.

Customizing Captive Portal

Customizing the following Captive Portal Web page components is allowed. These components are incorporated and displayed when the Web-based login page is presented to the user.

- Logo
- Welcome text
- Background image
- User Acceptable Policy text
- Login help page

To create a custom version of any of the above components, create one or more of the following file types:

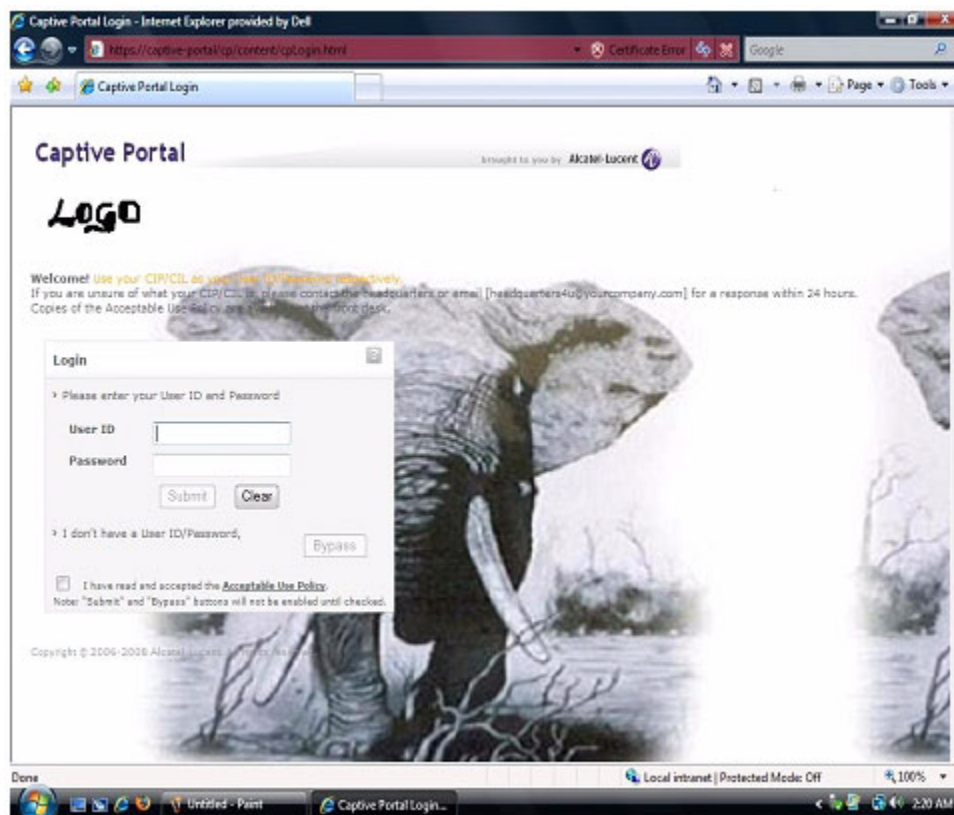
- **logo.gif, logo.jpg, or logo.png**—Use these files to provide a company logo that Captive Portal will display on all pages.
- **background.gif, background.jpg, or background.png**—Use these files to provide a page background image that Captive Portal will display on all pages.
- **cpPolicy.html**—The User Acceptable Policy HTML file that is linked to the Captive Portal login page. The link provided opens a new browser window to display the policy information.
- **cpLoginWelcome.inc, cpStatusWelcome.inc, cpFailWelcome.inc, cpBypassWelcome.inc**—Use these files to customize the welcome message for the Captive Portal login, successful status, fail status, and bypass status page.
- **cpLoginHelp.html**—Use this file to customize the Captive Portal login help page. A question-mark (“?”) button links to this HTML help page, which is displayed in a separate browser window.

Once the custom files are created with the images and information the file type requires, download the files to the **/flash/switch** directory on the switch. When a Captive Portal session is initiated, the switch checks to see if there are any files in this directory; if so, then the custom files are incorporated and displayed by Captive Portal. If no files are found, the default Captive Portal Web page components are used.

Consider the following guidelines when customizing Captive Portal Web page components:

- Filenames are case sensitive. When creating a custom file, make sure the filename matches the filename exactly as shown in the list of file types described above.
- Create custom logo and background pages using the **.gif**, **.jpg**, or **.png** formats. Captive Portal checks the **/flash/switch** directory on the switch for a **.gif** file, then a **.jpg** file, and finally a **.png** file. Whichever file type Captive Portal encounters first is the file used to display the custom logo or background.
- The **.inc** files, which are used to present customized welcome messages, are partial HTML files that can include only text or text and other HTML tags, such as links. Note that these **.inc** files are wrapped in a paragraph HTML tag within the body of a Captive Portal default page.

The following is an example of a customized Captive Portal login page:



Authenticating with Captive Portal

Access Guardian determines that a client device is a candidate for Web-based authentication if the following conditions are true:

- The device is connected to an 802.1x-enabled port.
- An Access Guardian policy (supplicant or non-supplicant) that includes the Captive Portal option is configured for the port.
- The device is not classified for VLAN assignment by any other policy or method configured for the port. For example, if a policy specifies Group Mobility and Captive Portal but device frames do not match any Group Mobility rules, then Access Guardian invokes Captive Portal authentication.

When all of the above conditions are met, Access Guardian places the device MAC address in a Captive Portal state. This means that the switch will not learn the device MAC address and a Web browser session is required to proceed with the authentication process.

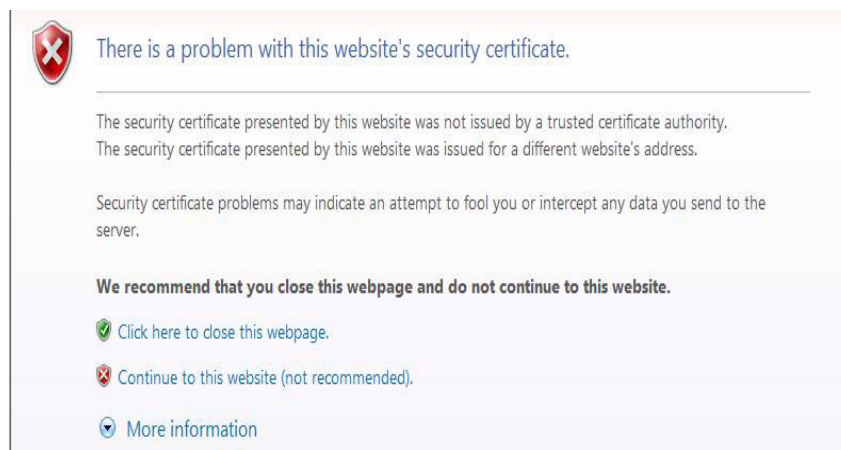
Note. Captive Portal does not require the configuration of IP interfaces, a UDP Relay agent, or an external DHCP server to provide an IP address for the client device. A temporary IP address derived from the Captive Portal subnet is assigned to the client for use during the authentication process. For more information, see [“Configuring Captive Portal Authentication” on page 30-22](#).

Logging Into the Network with Captive Portal

Once a user device is in the Captive Portal state, the following steps are required to complete the authentication process:

- 1 Open a Web browser window on the client device. If there is a default home page, the browser will attempt to connect to that URL. If a default home page is not available, enter a URL for any website and attempt to connect to that site. Note that the specified URL must contain the “http”, “https”, or “www” prefix (see [“Configuring Captive Portal Authentication” on page 30-22](#) for more information).

A certificate warning message may appear when the Web browser window opens. If so, select the option to continue on to the website. For example, Windows IE7 browser displays the following message:



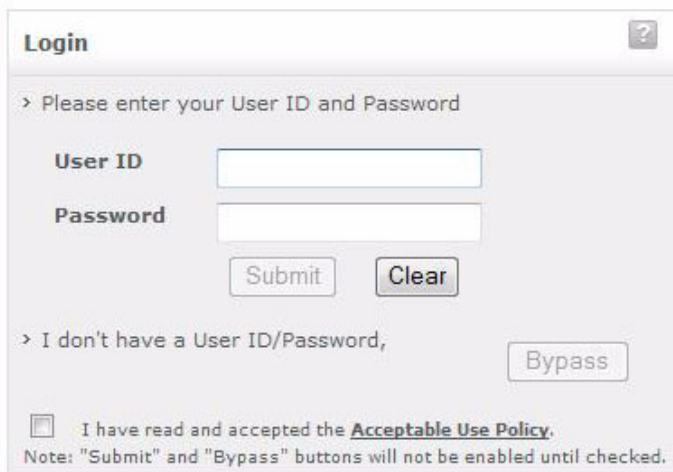
When the browser window opens and after the certificate warning message, if any, is cleared, Captive Portal displays a login screen similar to the one shown in the following example:

Captive Portal

brought to you by Alcatel-Lucent 

Welcome! Use your CIP/CIL as your User ID/Password respectively.

If you are unsure of what your CIP/CIL is, please contact the headquarters or email [headquarters4u@yourcompany.c
Copies of the Acceptable Use Policy are available at the front desk.



Login ?

> Please enter your User ID and Password

User ID

Password

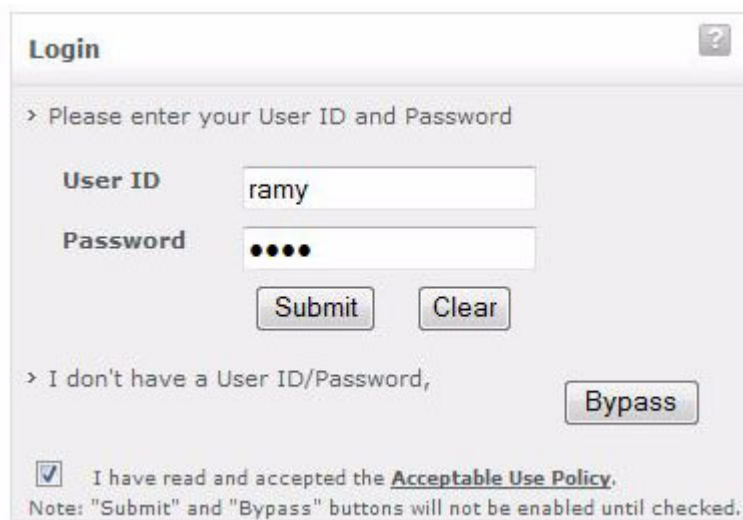
> I don't have a User ID/Password,

I have read and accepted the [Acceptable Use Policy](#).

Note: "Submit" and "Bypass" buttons will not be enabled until checked.

Copyright © 2006-2009 Alcatel-Lucent. All rights reserved.

- 2 Enter the user name in the “User ID” field.
- 3 Enter the user password in the “Password” field.
- 4 Click on the “Acceptable Use Policy” box to activate the “Submit” and “Bypass” buttons, as shown below:



Login ?

> Please enter your User ID and Password

User ID

Password

> I don't have a User ID/Password,

I have read and accepted the [Acceptable Use Policy](#).

Note: "Submit" and "Bypass" buttons will not be enabled until checked.

5 Click the “Submit” button to login to the network or click the “Bypass” button to bypass Captive Portal authentication (see [“Bypassing Captive Portal Login” on page 30-27](#)). If the “Submit” button is clicked, Captive Portal sends the user information provided in the login window to the RADIUS server for authentication. The following status message appears during the authentication process:



6 If user authentication is successful, the following status and logout messages are displayed:



The user is now logged into the network and has access to all network resources in the VLAN to which this user was assigned. The VLAN membership for the user was either returned through RADIUS authentication or determined through Captive Portal device classification (invoked when RADIUS does not return a VLAN ID or authentication fails).

7 Click on “Bookmark the CP-Logout link” or make note of the “http://captive-portal/logout” URL before leaving the Captive Portal status page or closing the browser window. See [“Logging Off the Network with Captive Portal” on page 30-28](#) for more information.

Note. The “http://captive-portal/logout” URL is used to display a Captive Portal logout page. If a user does not log out of a Captive Portal session using this URL, the session remains active until the Captive Portal session limit is reached (default is 12 hours). Adding a bookmark for this URL is highly recommended.

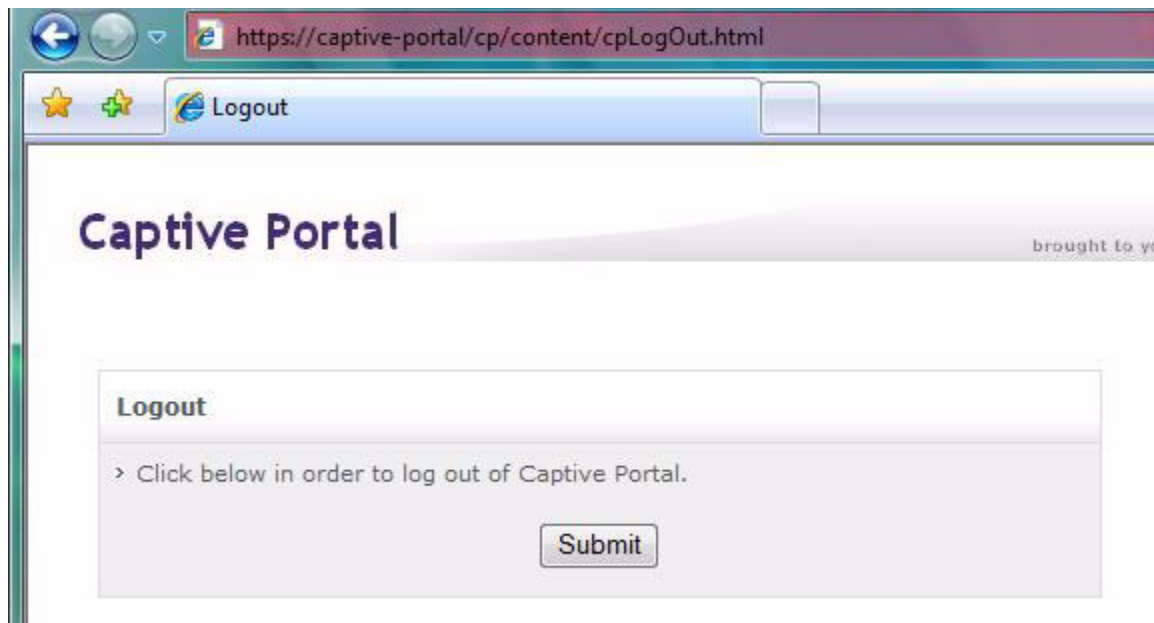
Bypassing Captive Portal Login

The Captive Portal login screen includes a “Bypass” button for users that do not have user credentials. When this option is selected, the authentication process is bypassed and the Captive Portal fail policy configured for the 802.1x port is applied to classify the device.

For more information about the Captive Portal policy, see [“Configuring the Captive Portal Policy” on page 30-20](#).

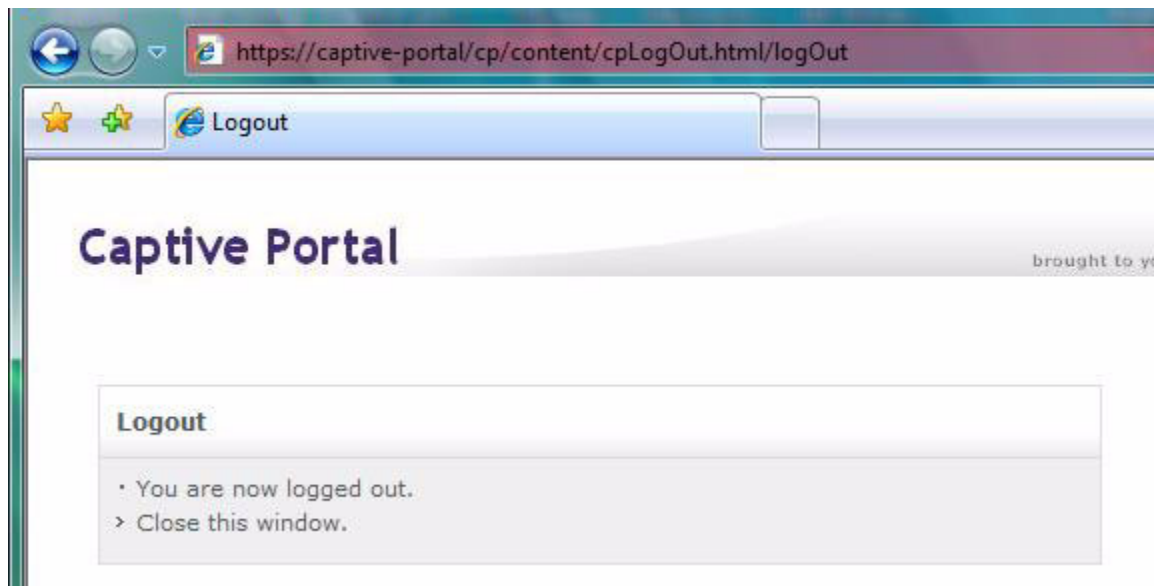
Logging Off the Network with Captive Portal

When “http://captive-portal/logout” URL is entered in the location bar of the browser or the URL bookmark is selected, the following Captive Portal logout page is displayed:



To log off from a Captive Portal session, the user clicks on the “Submit” button. The user is then logged off the network and the user device returns to the Captive Portal state (device MAC address is unknown to the switch).

The following logout confirmation page appears when the logout process is done.



Note. A user is automatically logged out of the network if the Captive Portal session time limit is reached. For more information, see [“Configuring Captive Portal Session Parameters”](#) on page 30-23.

Configuring User Network Profiles

User Network Profiles (UNP) are applied through 802.1X or MAC-based authentication. Upon successful authentication, the RADIUS server returns a UNP name (contained in the “Filter-ID” attribute) that is used to classify the authenticated device.

The UNP name returned by the RADIUS server must match an existing UNP name configured for the switch. The switch UNP specifies a VLAN ID assignment that is applied to all authenticated devices to which the UNP name is assigned through authentication. If a device fails authentication or there is not a matching UNP name configured on the switch, the device is blocked from accessing the network.

To configure a UNP, use the **aaa user-network-profile** command. For example, the following command creates the “guest_user” profile to assign devices to VLAN 500:

```
-> aaa user-network-profile name guest_user vlan 500
```

Note that assigning a UNP is not a configurable option for device classification policies. Using profiles to classify user devices is configured by specifying a UNP name in the RADIUS server attribute and configuring a UNP on the switch with the same profile name.

To verify the UNP configuration for the switch, use the **show aaa user-network-profile** command. For more information about user profiles, see [“User Network Profiles \(Role-Based Access\)” on page 30-10](#). For more information about configuring RADIUS servers, see [Chapter 31, “Managing Authentication Servers.”](#)

Verifying the Access Guardian Configuration

A summary of the **show** commands used for verifying the Access Guardian configuration is given here:

802.1x auth-server-down	Displays information about ports configured for 802.1X. Includes Captive Portal session timeout and login retry parameter values.
show 802.1x auth-server-down	Displays global information about the Access Guardian Captive Portal configuration.
show 802.1x device classification policies	Displays Access Guardian device classification policies configured for 802.1x-enabled ports.
show aaa user-network-profile	Displays the User Network Profile (UNP) configuration for the switch.
show aaa priv hexa	Displays the global Host Integrity Check (HIC) configuration for the switch.
show aaa authentication 802.1x	Displays information about the global 802.1X configuration on the switch.
show aaa authentication mac	Displays a list of RADIUS servers configured for MAC based authentication.

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

31 Managing Authentication Servers

This chapter describes authentication servers and how they are used with the switch. The types of servers described include Remote Authentication Dial-In User Service (RADIUS), Lightweight Directory Access Protocol (LDAP), Terminal Access Controller Access Control System (TACACS+), and SecurID's ACE/Server.

In This Chapter

The chapter includes some information about attributes that must be configured on the servers, but it primarily addresses configuring the switch through the Command Line Interface (CLI) to communicate with the servers to retrieve authentication information about users.

Configuration procedures described include:

- **Configuring an ACE/Server.** This procedure is described in [“ACE/Server” on page 31-7](#).
- **Configuring a RADIUS Server.** This procedure is described in [“RADIUS Servers” on page 31-8](#).
- **Configuring a TACACS+ Server.** This procedure is described in [“TACACS+ Server” on page 31-14](#).
- **Configuring an LDAP Server.** This procedure is described in [“LDAP Servers” on page 31-16](#).

For information about using servers for authenticating users to manage the switch, see the “Switch Security” chapter in the *OmniSwitch 6250 Switch Management Guide*.

Authentication Server Specifications

RADIUS RFCs Supported	<p>RFC 2865–Remote Authentication Dial In User Service (RADIUS)</p> <p>RFC 2866–RADIUS Accounting</p> <p>RFC 2867–RADIUS Accounting Modifications for Tunnel Protocol Support</p> <p>RFC 2868–RADIUS Attributes for Tunnel Protocol Support</p> <p>RFC 2809–Implementation of L2TP Compulsory Tunneling via RADIUS</p> <p>RFC 2869–RADIUS Extensions</p> <p>RFC 2548–Microsoft Vendor-specific RADIUS Attributes</p> <p>RFC 2882–Network Access Servers Requirements: Extended RADIUS Practices</p>
TACACS+ RFCs Supported	RFC 1492–An Access Control Protocol
LDAP RFCs Supported	<p>RFC 1789–Connectionless Lightweight X.5000 Directory Access Protocol</p> <p>RFC 2247–Using Domains in LDAP/X.500 Distinguished Names</p> <p>RFC 2251–Lightweight Directory Access Protocol (v3)</p> <p>RFC 2252–Lightweight Directory Access Protocol (v3): Attribute Syntax Definitions</p> <p>RFC 2253–Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names</p> <p>RFC 2254–The String Representation of LDAP Search Filters</p> <p>RFC 2256–A Summary of the X.500(96) User Schema for Use with LDAPv3</p>
Other RFCs	<p>RFC 2574–User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)</p> <p>RFC 2924–Accounting Attributes and Record Formats</p> <p>RFC 2975–Introduction to Accounting Management</p> <p>RFC 2989–Criteria for Evaluating AAA Protocols for Network Access</p>
Platforms Supported	OmniSwitch 6250 Series
Maximum number of authentication servers in single authority mode	4 (not including any backup servers)
Maximum number of authentication servers in multiple authority mode	4 per VLAN (not including any backup servers)
Maximum number of servers per Authenticated Switch Access type	4 (not including any backup servers)
CLI Command Prefix Recognition	The aaa radius-server , aaa tacacs+-server , and aaa ldap-server commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

Server Defaults

The defaults for authentication server configuration on the switch are listed in the tables in the next sections.

RADIUS Authentication Servers

Defaults for the `aaa radius-server` command are as follows:

Description	Keyword	Default
Number of retries on the server before the switch tries a backup server	<code>retransmit</code>	3
Timeout for server replies to authentication requests	<code>timeout</code>	2
UDP destination port for authentication	<code>auth-port</code>	1645*
UDP destination port for accounting	<code>acct-port</code>	1646*

* The port defaults are based on the older RADIUS standards; some servers are set up with port numbers based on the newer standards (ports 1812 and 1813, respectively).

TACACS+ Authentication Servers

Defaults for the `aaa tacacs+-server` command are as follows:

Description	Keyword	Default
Timeout for server replies to authentication requests	<code>timeout</code>	2
The port number for the server	<code>port</code>	49

LDAP Authentication Servers

Defaults for the `aaa ldap-server` command are as follows:

Description	Keyword	Default
The port number for the server	<code>port</code>	389 (SSL disabled) 636 (SSL enabled)
Number of retries on the server before the switch tries a backup server	<code>retransmit</code>	3
Timeout for server replies to authentication requests	<code>timeout</code>	2
Whether a Secure Socket Layer is configured for the server	<code>ssl</code> <code>no ssl</code>	<code>no ssl</code>

Quick Steps For Configuring Authentication Servers

- 1 For RADIUS, TACACS+, or LDAP servers, configure user attribute information on the servers. See [“RADIUS Servers” on page 31-8](#), [“TACACS+ Server” on page 31-14](#), and [“LDAP Servers” on page 31-16](#).
- 2 Use the **aaa radius-server**, **aaa tacacs+-server**, and/or the **aaa ldap-server** command to configure the authentication server(s). For example:

```
-> aaa radius-server rad1 host 10.10.2.1 10.10.3.5 key amadeus
-> aaa tacacs+-server tac3 host 10.10.4.2 key otna timeout 10
-> aaa ldap-server ldap2 host 10.10.3.4 dn cn=manager password tpub base c=us
```

Note. (Optional) Verify the server configuration by entering the **show aaa server** command. For example:

```
-> show aaa server
Server name = rad1
  Server type           = RADIUS,
  IP Address 1         = 10.10.2.1,
  IP Address 2         = 10.10.3.5
  Retry number         = 3,
  Timeout (in sec)    = 2,
  Authentication port  = 1645,
  Accounting port      = 1646
Server name = ldap2
  Server type           = LDAP,
  IP Address 1         = 10.10.3.4,
  Port                 = 389,
  Domain name          = cn=manager,
  Search base          = c=us,
  Retry number         = 3,
  Timeout (in sec)    = 2,
Server name = Tacacs1
  ServerIp             = 1.1.1.1
  ServerPort           = 49
  Encryption           = MD5
  Timeout              = 5 seconds
  Status               = UP
```

See the *CLI Reference Guide* for information about the fields in this display.

- 3 If you are using ACE/Server, there is no required switch configuration; however, you must FTP the **sdconf.rec** file from the server to the switch's **/network** directory.
- 4 Configure authentication on the switch. This step is described in other chapters. For a quick overview of using the configured authentication servers for 802.1X and MAC-based authentication, see [Chapter 32, “Configuring 802.1X.”](#) For a quick overview of using the configured authentication servers with Authenticated Switch Access, see the *OmniSwitch 6250 Switch Management Guide*.

Server Overview

Authentication servers are sometimes referred to as AAA servers (authentication, authorization, and accounting). These servers are used for storing information about users who want to manage the switch (Authenticated Switch Access) and users who need access to a particular VLAN or VLANs.

RADIUS, TACACS+, LDAP, and SecurID's ACE/Server may be used for Authenticated Switch Access. However, only RADIUS servers are supported for 802.1X Port-based Network Access Control.

The following table describes how each type of server may be used with the switch:

Server Type	Authenticated Switch Access	802.1X Port-Based Network Access Control
ACE/Server	yes (except SNMP)	no
RADIUS	yes (except SNMP)	yes
TACACS+	yes (including SNMP)	no
LDAP	yes (including SNMP)	no

Backup Authentication Servers

Each RADIUS, TACACS+, and LDAP server may have one backup host (of the same type) configured through the [aaa radius-server](#), [aaa tacacs+-server](#), and [aaa ldap-server](#) commands, respectively. In addition, each authentication method (Authenticated Switch Access, or 802.1X) may specify a list of backup authentication servers that includes servers of different types (if supported on the feature).

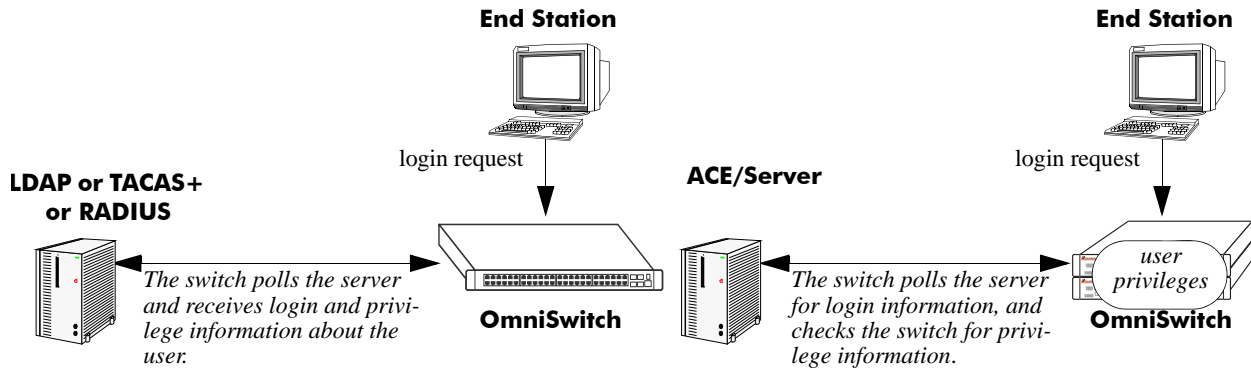
The switch uses the first available authentication server to attempt to authenticate users. If user information is not found on the first available server, the authentication attempts fails.

Authenticated Switch Access

When RADIUS, TACACS+, and/or LDAP servers are set up for Authenticated Switch Access, the switch polls the server for user login information. The switch also polls the server for privilege information (authorization) if it has been configured on the server; otherwise, the local user database is polled for the privileges.

For RADIUS, TACACS+, and LDAP, additional servers may be configured as backups.

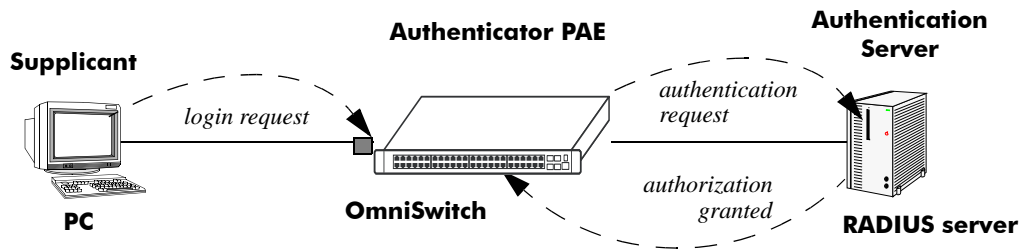
A RADIUS server supporting the challenge and response mechanism as defined in RADIUS RFC 2865 may access an ACE/Server for authentication purposes. The ACE/Server is then used for user authentication, and the RADIUS server is used for user authorization.



Servers Used for Authenticated Switch Access

Port-Based Network Access Control (802.1X)

For devices authenticating on an 802.1X port on the switch, only RADIUS authentication servers are supported. The RADIUS server contains a database of user names and passwords, and may also contain challenges/responses and other authentication criteria.



Basic 802.1X Components

For more information about configuring 802.1X ports on the switch, see [Chapter 32, “Configuring 802.1X.”](#)

ACE/Server

An external ACE/Server may be used for authenticated switch access. It cannot be used for Layer 2 authentication or for policy management. Attributes are not supported on ACE/Servers. These values must be configured on the switch through the **user** commands. See the “Switch Security” chapter of the *OmniSwitch 6250 Switch Management Guide* for more information about setting up the local user database.

Since an ACE/Server does not store or send user privilege information to the switch, user privileges for SecurID logins are determined by the switch. When a user attempts to log into the switch, the user ID and password is sent to the ACE/Server. The server determines whether the login is valid. If the login is valid, the user privileges must be determined. The switch checks its user database for the user’s privileges. If the user is not in the database, the switch uses the default privilege, which is determined by the default user account. For information about the default user account, see the “Switch Security” chapter of the *OmniSwitch 6250 Switch Management Guide*.

There are no server-specific parameters that must be configured for the switch to communicate with an attached ACE/Server; however, you must FTP the **sdconf.rec** file from the server to the switch’s **/network** directory. This file is required so that the switch will know the IP address of the ACE/Server. For information about loading files onto the switch, see the *OmniSwitch 6250 Switch Management Guide*.

The ACE client in the switch is version 4.1; it does not support the replicating and locking feature of ACE 5.0, but it may be used with an ACE 5.0 server if a legacy configuration file is loaded on the server. The legacy configuration must specify authentication to two specific servers (master and slave). See the RSA Security ACE/Server documentation for more information.

To display information about any servers configured for authentication, use the **show aaa server** command. For more information about the output for this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Also, you may need to clear the ACE/Server secret occasionally because of misconfiguration or required changes in configuration. Clearing the secret is described in the next section.

Clearing an ACE/Server Secret

The ACE/Server generates “secrets” that it sends to clients for authentication. While you cannot configure the secret on the switch, you can clear it. The secret may need to be cleared because the server and the switch get out of sync. See the RSA Security ACE/Server documentation for more information about the server secret.

To clear the secret on the switch, enter the following command:

```
-> aaa ace-server clear
```

When you clear the secret on the switch, the secret must also be cleared on the ACE/Server as described by the RSA Security ACE/Server documentation.

RADIUS Servers

RADIUS is a standard authentication and accounting protocol defined in RFC 2865 and RFC 2866. A built-in RADIUS client is available in the switch. A RADIUS server that supports Vendor Specific Attributes (VSAs) is required. The Alcatel-Lucent attributes may include VLAN information, time-of-day, or slot/port restrictions.

RADIUS Server Attributes

RADIUS servers and RADIUS accounting servers are configured with particular attributes defined in RFC 2138 and RFC 2139, respectively. These attributes carry specific authentication, authorization, and configuration details about RADIUS requests to and replies from the server. This section describes the attributes and how to configure them on the server.

Standard Attributes

The following tables list RADIUS server attributes 1–39 and 60–63, their descriptions, and whether the Alcatel-Lucent RADIUS client in the switch supports them. Attribute 26 is for vendor-specific information and is discussed in [“Vendor-Specific Attributes for RADIUS” on page 31-10](#). Attributes 40–59 are used for RADIUS accounting servers and are listed in [“RADIUS Accounting Server Attributes” on page 31-11](#).

Num.	Standard Attribute	Notes
1	User-Name	Used in access-request and account-request packets.
2	User-Password	—
3	CHAP-Password	<i>Not supported.</i>
4	NAS-IP-Address	Sent with every access-request. Specifies which switches a user may have access to. More than one of these attributes is allowed per user.
5	NAS-Port	Virtual port number sent with access-request and account-request packets. Slot/port information is supplied in attribute 26 (vendor-specific).
6	Service-Type	<i>Not supported. These attributes are used for dial-up sessions; not applicable to the RADIUS client in the switch.</i>
7	Framed-Protocol	
8	Framed-IP-Address	
9	Framed-IP-Netmask	
10	Framed-Routing	
11	Filter-Id	Used to return a User Network Profile (UNP) name.
12	Framed-MTU	<i>Not supported. These attributes are used for dial-up sessions; not applicable to the RADIUS client in the switch.</i>
13	Framed-Compression	
14	Login-IP-Host	
15	Login-Service	
16	Login-TCP-Port	
17	Unassigned	—
18	Reply-Message	Multiple reply messages are supported, but the length of all the reply messages returned in one access-accept or access-reject packet cannot exceed 256 characters.

Num.	Standard Attribute	Notes
19	Callback-Number	<i>Not supported. These attributes are used for dial-up sessions; not applicable to the RADIUS client in the switch.</i>
20	Callback-Id	
21	Unassigned	
22	Frame-Route	
23	Framed-IPX-Network	
24	State	Sent in challenge/response packets.
25	Class	Used to pass information from the server to the client and passed unchanged to the accounting server as part of the accounting-request packet.
26	Vendor-Specific	See “Vendor-Specific Attributes for RADIUS” on page 31-10.
27	Session-Timeout	<i>Not supported.</i>
28	Idle-Timeout	<i>Not supported.</i>
29	Termination-Action	<i>Not supported. These attributes are used for dial-up sessions; not applicable to the RADIUS client in the switch.</i>
30	Called-Station-Id	
31	Calling-Station-Id	
32	NAS-Identifier	
33	Proxy-State	
34	Login-LAT-Service	
35	Login-LAT-Node	
36	Login-LAT-Group	
37	Framed-AppleTalk-Link	
38	Framed-AppleTalk-Network	
39	Framed-AppleTalk-Zone	
60	CHAP-Challenge	
61	NAS-Port-Type	
62	Port-Limit	
63	Login-LAT-Port	

Vendor-Specific Attributes for RADIUS

The Alcatel-Lucent RADIUS client supports attribute 26, which includes a vendor ID and some additional sub-attributes called subtypes. The vendor ID and the subtypes collectively are called Vendor Specific Attributes (VSAs). Alcatel-Lucent, through partnering arrangements, has included these VSAs in some vendors' RADIUS server configurations.

The attribute subtypes are defined in the server's dictionary file. Alcatel-Lucent's vendor ID is 800 (SMI Network Management Private Enterprise Code).

The following are VSAs for RADIUS servers:

Num.	RADIUS VSA	Type	Description
1	Alcatel-Lucent-Auth-Group	integer	The VLAN number. The only protocol associated with this attribute is Ethernet II. If other protocols are required, use the protocol attribute instead.
2	Alcatel-Lucent-Slot-Port	string	Slot(s)/port(s) valid for the user.
3	Alcatel-Lucent-Time-of-Day	string	The time of day valid for the user to authenticate.
4	Alcatel-Lucent-Client-IP-Addr	address	The IP address used for Telnet only.
5	Alcatel-Lucent-Group-Desc	string	Description of the VLAN.
6	Alcatel-Lucent-Port-Desc	string	Description of the port.
8	Alcatel-Lucent-Auth-Group-Protocol	string	The protocol associated with the VLAN. Must be configured for access to other protocols. Values include: IP_E2 , IP_SNAP .
9	Alcatel-Lucent-Asa-Access	string	Specifies that the user has access to the switch. The only valid value is all .
39	Alcatel-Lucent-Acce-Priv-F-R1	hex.	Configures functional read privileges for the user.
40	Alcatel-Lucent-Acce-Priv-F-R2	hex.	Configures functional read privileges for the user.
41	Alcatel-Lucent-Acce-Priv-F-W1	hex.	Configures functional write privileges for the user.
42	Alcatel-Lucent-Acce-Priv-F-W2	hex.	Configures functional write privileges for the user.

The Alcatel-Lucent-Auth-Group attribute is used for Ethernet II only. If a different protocol, or more than one protocol is required, use the Alcatel-Lucent-Auth-Group-Protocol attribute instead. For example:

```
Alcatel-Lucent-Auth-Group-Protocol 23: IP_E2 IP_SNAP
```

In this example, authenticated users on VLAN 23 may use Ethernet II or SNAP encapsulation. .

Configuring Functional Privileges on the Server

Configuring the functional privileges attributes (**Alcatel-Lucent-Acce-Priv-F-x**) can be cumbersome because it requires using read and write bitmasks for command families on the switch.

- 1 To display the functional bitmasks of the desired command families, use the **show aaa priv hexa** command.
- 2 On the RADIUS server, configure the functional privilege attributes with the bitmask values.

Note. For more information about configuring users on the switch, see the “Switch Security” chapter in the *OmniSwitch 6250 Switch Management Guide*.

RADIUS Accounting Server Attributes

The following table lists the standard attributes supported for RADIUS accounting servers. The attributes in the **radius.ini** file may be modified if necessary.

Num.	Standard Attribute	Description
1	User-Name	Used in access-request and account-request packets.
4	NAS-IP-Address	Sent with every access-request. Specifies which switches a user may have access to. More than one of these attributes is allowed per user.
5	NAS-Port	Virtual port number sent with access-request and account-request packets. Slot/port information is supplied in attribute 26 (vendor-specific).
25	Class	Used to pass information from the server to the client and passed unchanged to the accounting server as part of the accounting-request packet.
40	Acct-Status-Type	Four values should be included in the dictionary file: 1 (acct-start), 2 (acct-stop), 6 (failure), and 7 (acct-on). Start and stop correspond to login/logout. The accounting-on message is sent when the RADIUS client is started. This attribute also includes an accounting-off value, which is not supported.
42	Acct-Input-Octets	Tracked per port.
43	Acct-Output-Octets	Tracked per port.
44	Acct-Session	Unique accounting ID.
45	Acct-Authentic	Indicates how the client is authenticated; standard values (1–3) are not used. Vendor specific values should be used instead: AUTH-AVCLIENT (4) AUTH-TELNET (5) AUTH-HTTP (6) AUTH-NONE (0)
46	Acct-Session	The start and stop time for a user’s session can be determined from the accounting log.
47	Acct-Input-Packets	Tracked per port.
48	Acct-Output-Packets	Tracked per port.

Num.	Standard Attribute	Description
49	Acct-Terminal-Cause	Indicates how the session was terminated: NAS-ERROR USER-ERROR LOST CARRIER USER-REQUEST STATUS-FAIL

The following table lists the VSAs supported for RADIUS accounting servers. The attributes in the **radius.ini** file may be modified if necessary.

Num.	Accounting VSA	Type	Description
1	Alcatel-Lucent-Auth-Group	integer	The VLAN number. The only protocol associated with this attribute is Ethernet II. If other protocols are required, use the protocol attribute instead.
2	Alcatel-Lucent-Slot-Port	string	Slot(s)/port(s) valid for the user.
4	Alcatel-Lucent-Client-IP-Addr	dotted decimal	The IP address used for Telnet only.
5	Alcatel-Lucent-Group-Desc	string	Description of the VLAN.

Configuring the RADIUS Client

Use the [aaa radius-server](#) command to configure RADIUS parameters on the switch.

RADIUS server keywords

key	timeout
host	auth-port
retransmit	acct-port

When creating a new server, at least one host name or IP address (specified by the **host** keyword) is required as well as the shared secret (specified by the **key** keyword).

In this example, the server name is **rad1**, the host address is 10.10.2.1, the backup address is 10.10.3.5, and the shared secret is **amadeus**. Note that the shared secret must be configured exactly the same as on the server.

```
-> aaa radius-server rad1 host 10.10.2.1 10.10.3.5 key amadeus
```

To modify a RADIUS server, enter the server name and the desired parameter to be modified.

```
-> aaa radius-server rad1 key mozart
```

If you are modifying the server and have just entered the **aaa radius-server** command to create or modify the server, you can use command prefix recognition. For example:

```
-> aaa radius-server rad1 retransmit 5
-> timeout 5
```

For information about server defaults, see [“Server Defaults” on page 31-3](#).

To remove a RADIUS server, use the **no** form of the command:

```
-> no aaa radius-server rad1
```

Note that only one server may be deleted at a time.

TACACS+ Server

Terminal Access Controller Access Control System (TACACS+) is a standard authentication and accounting protocol defined in RFC 1321 that employs TCP for reliable transport. A built-in TACACS+ client is available in the switch. A TACACS+ server allows access control for routers, network access servers, and other networked devices through one or more centralized servers. The protocol also allows separate authentication, authorization, and accounting services. By allowing arbitrary length and content authentication exchanges, it allows clients to use any authentication mechanism.

The TACACS+ client offers the ability to configure multiple TACACS+ servers. This can be done by the user. When the primary server fails, the client tries the subsequent servers. Multiple server configurations are applicable only for backup and not for server chaining.

In the TACACS+ protocol, the client queries the TACACS+ server by sending TACACS+ requests. The server responds with reply packets indicating the status of the request.

- **Authentication.** TACACS+ protocol provides authentication between the client and the server. It also ensures confidentiality because all the exchanges are encrypted. The protocol supports fixed passwords, one-time passwords, and challenge-response queries. Authentication is not a mandatory feature, and it can be enabled without authorization and accounting. During authentication if a user is not found on the primary TACACS+ server, the authentication fails. The client does not try to authenticate with the other servers in a multiple server configuration. If the authentication succeeds, then Authorization is performed.
- **Authorization.** Enabling authorization determines if the user has the authority to execute a specified command. TACACS+ authorization cannot be enabled independently. The TACACS+ authorization is enabled automatically when the TACACS+ authentication is enabled.
- **Accounting.** The process of recording what the user is attempting to do or what the user has done is Accounting. The TACACS+ accounting must be enabled on the switches for accounting to succeed. Accounting can be enabled irrespective of authentication and authorization. TACACS+ supports three types of accounting:

Start Records—Indicate the service is about to begin.

Stop Records—Indicates the services has just terminated.

Update Records—Indicates the services are still being performed.

TACACS+ Client Limitations

The following limitation apply to this implementation of the TACACS+ client application:

- TACACS+ supports Authenticated Switch Access and cannot be used for user authentication.
- Authentication and Authorization are combined together and cannot be performed independently.
- On the fly, command authorization will not be supported. Authorization will be similar to the AOS partition management families.
- Only inbound ASCII logins are supported.
- A maximum of 50 simultaneous TACACS+ sessions can be supported when no other authentication mechanism is activated.
- Accounting of commands performed by the user on the remote TACACS+ process will not be supported at in the boot.cfg file at boot up time.

Configuring the TACACS+ Client

Use the `aaa tacacs+-server` command to configure TACACS+ parameters on the switch.

TACACS+ server keywords

key	timeout
host	port

When creating a new server, at least one host name or IP address (specified by the **host** keyword) is required as well as the shared secret (specified by the **key** keyword).

In this example, the server name is **tacl**, the host address is 10.10.5.2, the backup address is 10.10.5.5, and the shared secret is **otna**. Note that the shared secret must be configured exactly the same as on the server.

```
-> aaa tacacs+-server tacl host 10.10.5.2 10.10.5.5 key otna
```

To modify a TACACS+ server, enter the server name and the desired parameter to be modified.

```
-> aaa tacacs+-server tacl key tnmelc
```

If you are modifying the server and have just entered the `aaa tacacs+-server` command to create or modify the server, you can use command prefix recognition. For example:

```
-> aaa tacacs+-server tacl timeout 5
```

For information about server defaults, see [“Server Defaults” on page 31-3](#).

To remove a TACACS+ server, use the **no** form of the command:

```
-> no aaa tacacs+-server tacl
```

Note that only one server may be deleted at a time.

LDAP Servers

Lightweight Directory Access Protocol (LDAP) is a standard directory server protocol. The LDAP client in the switch is based on several RFCs: 1798, 2247, 2251, 2252, 2253, 2254, 2255, and 2256. The protocol was developed as a way to use directory services over TCP/IP and to simplify the directory access protocol (DAP) defined as part of the Open Systems Interconnection (OSI) effort. Originally it was a front-end for X.500 DAP.

The protocol synchronizes and governs the communications between the LDAP client and the LDAP server. The protocol also dictates how its databases of information, which are normally stored in hierarchical form, are searched, from the root directory down to distinct entries.

In addition, LDAP has its own format that permits LDAP-enabled Web browsers to perform directory searches over TCP/IP.

Setting Up the LDAP Authentication Server

- 1 Install the directory server software on the server.
- 2 Copy the relevant schema LDIF files from the Alcatel-Lucent software CD to the configuration directory on the server. (Each server type has a command line tool or a GUI tool for importing LDIF files.) Database LDIF files may also be copied and used as templates. The schema files and the database files are specific to the server type. The files available on the Alcatel-Lucent software CD include the following:

```
aaa_schema.microsoft.ldif
aaa_schema.netscape.ldif
aaa_schema.novell.ldif
aaa_schema.openldap.schema
aaa_schema.sun.ldif

aaa_database.microsoft.ldif
aaa_database.netscape.ldif
aaa_database.novell.ldif
aaa_database.openldap.ldif
aaa_database.sun.ldif
```

- 3 After the server files have been imported, restart the server.

Note. Schema checking should be enabled on the server.

Information in the server files must match information configured on the switch through the **aaa ldap-server** command. For example, the port number configured on the server must be the same as the port number configured on the switch. See [“Configuring the LDAP Authentication Client” on page 31-26](#) for information about using this command.

LDAP Server Details

LDAP servers must be configured with the properly defined LDAP schema and correct database suffix, including well-populated data. LDAP schema is extensible, permitting entry of user-defined schema as needed.

LDAP servers are also able to import and export directory databases using LDIF (LDAP Data Interchange Format).

LDIF File Structure

LDIF is used to transfer data to LDAP servers in order to build directories or modify LDAP databases. LDIF files specify multiple directory entries or changes to multiple entries, but not both. The file is in simple text format and can be created or modified in any text editor. In addition, LDIF files import and export binary data encoded according to the base 64 convention used with MIME (Multipurpose Internet Mail Extensions) to send various media file types, such as JPEG graphics, through electronic mail.

An LDIF file entry used to define an organizational unit would look like this:

```
dn: <distinguished name>
objectClass: top
objectClass: organizationalUnit
ou: <organizational unit name>
<list of optional attributes>
```

Below are definitions of some LDIF file entries:

entries	definition
dn: <distinguished name>	Defines the DN (required).
objectClass: top	Defines top object class (at least one is required). Object class defines the list of attributes required and allowed in directory server entries.
objectClass: organizationalUnit	Specifies that organizational unit should be part of the object class.
ou: <organizationalUnit name>	Defines the organizational unit's name.
<list of attributes>	Defines the list of optional entry attributes.

Common Entries

The most common LDIF entries describe people in companies and organizations. The structure for such an entry might look like the following:

```
dn: <distinguished name>
objectClass: top
objectClass: person
objectClass: organizational Person
cn: <common name>
sn: <surname>
<list of optional attributes>
```

This is how the entry would appear with actual data in it.

```
dn: uid=yname, ou=people, o=yourcompany  
objectClass: top  
objectClass: person  
objectClass: organizational Person  
cn: your name  
sn: last name  
givenname: first name  
uid: yname  
ou: people  
description:  
<list of optional attributes>  
...
```

Directory Entries

Directory entries are used to store data in directory servers. LDAP-enabled directory entries contain information about an object (person, place, or thing) in the form of a Distinguished Name (DN) that should be created in compliance with the LDAP protocol naming conventions.

Distinguished names are constructed from Relative Distinguished Names (RDNs), related entries that share no more than one attribute value with a DN. RDNs are the components of DNs, and DNs are string representations of entry names in directory servers.

Distinguished names typically consist of descriptive information about the entries they name, and frequently include the full names of individuals in a network, their email addresses, TCP/IP addresses, with related attributes such as a department name, used to further distinguish the DN. Entries include one or more object classes, and often a number of attributes that are defined by values.

Object classes define all required and optional attributes (a set of object classes is referred to as a “schema”). As a minimum, every entry must include the DN and one defined object class, like the name of an organization. Attributes required by a particular object class must also be defined. Some commonly used attributes that comprise a DN include the following:

**Country (c), State or Province (st), Locality (l),
Organization (o), Organization Unit (ou),
and Common Name (cn)**

Although each attribute would necessarily have its own values, the attribute syntax determines what kind of values are allowed for a particular attribute, e.g., (c=US), where country is the attribute and US is the value. Extra consideration for attribute language codes will be necessary if entries are made in more than one language.

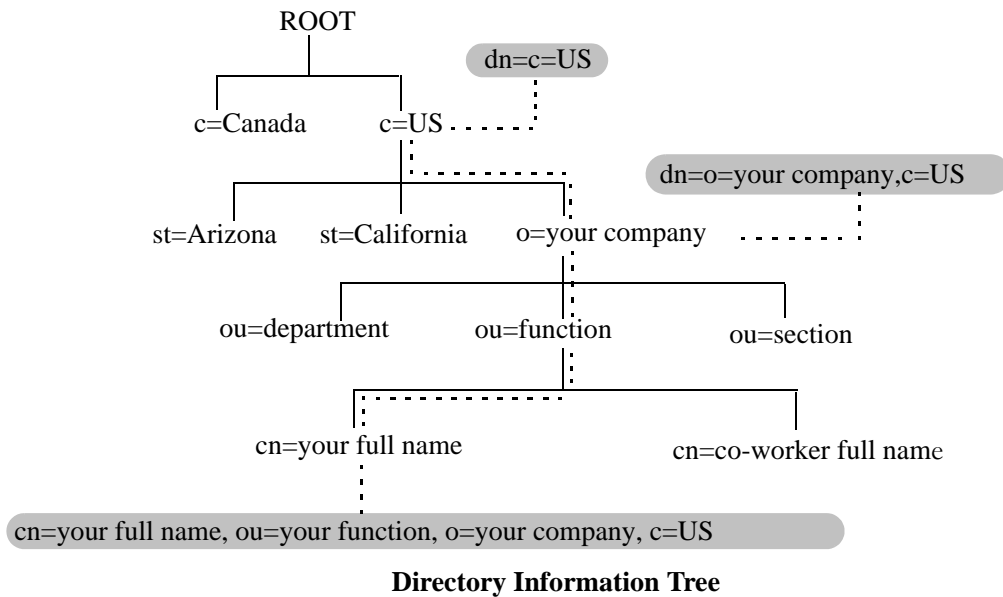
Entries are usually based on physical locations and established policies in a Directory Information Tree (DIT); the DN locates an entry in the hierarchy of the tree. Alias entries pointing to other entries can also be used to circumvent the hierarchy during searches for entries.

Once a directory is set up, DN attributes should thereafter be specified in the same order to keep the directory paths consistent. DN attributes are separated by commas as shown in this example:

cn=your name, ou=your function, o= your company, c=US

As there are other conventions used, please refer to the appropriate RFC specification for further details.

In addition to managing attributes in directory entries, LDAP makes the descriptive information stored in the entries accessible to other applications. The general structure of entries in a directory tree is shown in the following illustration. It also includes example entries at various branches in the tree.



Directory Searches

DNs are always the starting point for searches unless indicated otherwise in the directory schema.

Searches involve the use of various criteria including scopes and filters which must be predefined, and utility routines, such as Sort. Searches should be limited in scope to specific durations and areas of the directory. Some other parameters used to control LDAP searches include the size of the search and whether to include attributes associated with name searches.

Base objects and scopes are specified in the searches, and indicate where to search in the directory. Filters are used to specify entries to select in a given scope. The filters are used to test the existence of object class attributes, and enable LDAP to emulate a “read” of entry listings during the searches. All search preferences are implemented by means of a filter in the search. Filtered searches are based on some component of the DN.

Retrieving Directory Search Results

Results of directory searches are individually delivered to the LDAP client. LDAP referrals to other servers are not returned to the LDAP client, only results or errors. If referrals are issued, the server is responsible for them, although the LDAP client will retrieve results of asynchronous operations.

Directory Modifications

Modifications to directory entries contain changes to DN entry attribute values, and are submitted to the server by an LDAP client application. The LDAP-enabled directory server uses the DNs to find the entries to either add or modify their attribute values.

Attributes are automatically created for requests to add values if the attributes are not already contained in the entries.

All attributes are automatically deleted when requests to delete the last value of an attribute are submitted. Attributes can also be deleted by specifying delete value operations without attaching any values.

Modified attribute values are replaced with other given values by submitting replace requests to the server, which then translates and performs the requests.

Directory Compare and Sort

LDAP will compare directory entries with given attribute values to find the information it needs. The Compare function in LDAP uses a DN as the identity of an entry, and searches the directory with the type and value of an attribute. Compare is similar to the Search function, but simpler.

LDAP will also sort entries by their types and attributes. For the Sort function, there are essentially two methods of sorting through directory entries. One is to sort by entries where the DN (Distinguished Name) is the sort key. The other is to sort by attributes with multiple values.

The LDAP URL

LDAP URLs are used to send search requests to directory servers over TCP/IP on the internet, using the protocol prefix: **ldap://**. (Searches over SSL would use the same prefix with an “s” at the end, i.e., **ldaps://**.)

LDAP URLs are entered in the command line of any web browser, just as HTTP or FTP URLs are entered. When LDAP searches are initiated LDAP checks the validity of the LDAP URLs, parsing the various components contained within the URLs to process the searches. LDAP URLs can specify and implement complex or simple searches of a directory depending on what is submitted in the URLs. Searches performed directly with LDAP URLs are affected by the LDAP session parameters described above.

In the case of multiple directory servers, LDAP URLs are also used for referrals to other directory servers when a particular directory server does not contain any portion of requested IP address information. Search requests generated through LDAP URLs are not authenticated.

Searches are based on entries for attribute data pairs.

The syntax for TCP/IP LDAP URLs is as follows:

ldap://<hostname>:<port>/<base_dn>?attributes?<scope>?<filter>

An example might be:

ldap://ldap.company name.xxx/o=company name%inc./,c=US>
(base search including all attributes/object classes in scope).

LDAP URLs use the percent symbol to represent commas in the DN. The following table shows the basic components of LDAP URLs.

components	description
<ldap>	Specifies TCP/IP connection for LDAP protocol. (The <ldaps> prefix specifies SSL connection for LDAP protocol.)
<hostname>	Host name of directory server or computer, or its IP address (in dotted decimal format).
<port>	TCP/IP port number for directory server. If using TCP/IP and default port number (389), port need not be specified in the URL. SSL port number for directory server (default is 636).

components	description
<base_dn>	DN of directory entry where search is initiated.
<attributes>	Attributes to be returned for entry search results. All attributes are returned if search attributes are not specified.
<scope>	Different results are retrieved depending on the scopes associated with entry searches. “base” search: retrieves information about distinguished name as specified in URL. This is a <base_dn> search. Base searches are assumed when the scope is not designated. “one” (one-level) search: retrieves information about entries one level under distinguished name (<base_dn> as specified in the URL, excluding the base entry. “sub” (subtree) search: retrieves information about entries from all levels under the distinguished name (<base_dn>) as specified in the URL, including the base entry.
<filter>	Search filters are applied to entries within specified search scopes. Default filter objectClass=* is used when filters are not designated. (Automatic search filtering not yet available.)

Password Policies and Directory Servers

Password policies applied to user accounts vary slightly from one directory server to another. Normally, only the password changing policies can be set by users through the directory server graphical user interface (GUI). Other policies accessible only to Network Administrators through the directory server GUI may include one or more of the following operational parameters.

- Log-in Restrictions
- Change Password
- Check Password Syntax
- Password Minimum Length
- Send Expiration Warnings
- Password History
- Account Lockout
- Reset Password Failure Count
- LDAP Error Messages (e.g., Invalid Username/Password, Server Data Error, etc.)

For instructions on installing LDAP-enabled directory servers, refer to the vendor-specific instructions.

Directory Server Schema for LDAP Authentication

Object classes and attributes will need to be modified accordingly to include LDAP authentication in the network (object classes and attributes are used specifically here to map user account information contained in the directory servers).

- All LDAP-enabled directory servers require entry of an auxiliary objectClass:passwordObject for user password policy information.
- Another auxiliary objectClass: password policy is used by the directory server to apply the password policy for the entire server. There is only one entry of this object for the database server.

Note. Server schema extensions should be configured before the **aaa ldap-server** command is configured.

Vendor-Specific Attributes for LDAP Servers

The following are Vendor Specific Attributes (VSAs) for Authenticated Switch Access and/or Layer 2 Authentication:

attribute	description
bop-asa-func-priv-read-1	Read privileges for the user.
bop-asa-func-priv-read-2	Read privileges for the user.
bop-asa-func-priv-write-1	Write privileges for the user.
bop-asa-func-priv-write-2	Write privileges for the user.
bop-asa-allowed-access	Whether the user has access to configure the switch.
bop-asa-snmp-level-security	Whether the user may have SNMP access, and the type of SNMP protocol used.
bop-shakey	A key computed from the user password with the alp2key tool.
bop-md5key	A key computed from the user password with the alp2key tool.
allowedtime	The periods of time the user is allowed to log into the switch.
switchgroups	The VLAN ID and protocol (IP_E2 , IP_SNAP).

Configuring Functional Privileges on the Server

Configuring the functional privileges attributes (**bop-asa-func-priv-read-1**, **bop-asa-func-priv-read-2**, **bop-asa-func-priv-write-1**, **bop-asa-func-priv-write-2**) requires using read and write bitmasks for command families on the switch.

- 1 To display the functional bitmasks of the desired command families, use the **show aaa priv hexa** command.
- 2 On the LDAP server, configure the functional privilege attributes with the bitmask values.

For more information about configuring users on the switch, see the Switch Security chapter of the *OmniSwitch 6250 Switch Management Guide*.

Configuring Authentication Key Attributes

The alp2key tool is provided on the Alcatel-Lucent software CD for computing SNMP authentication keys. The alp2key application is supplied in two versions, one for Unix (Solaris 2.5.1 or higher) and one for Windows (NT 4.0 and higher).

To configure the bop-shakey or bop-md5key attributes on the server:

- 1 Use the alp2key application to calculate the authentication key from the password of the user. The switch automatically computes the authentication key, but for security reasons the key is never displayed in the CLI.
- 2 Cut and paste the key to the relevant attribute on the server.

An example using the alp2key tool to compute the SHA and MD5 keys for **mypassword**:

```
ors40595{}128: alp2key mypassword
bop-shakey: 0xb1112e3472ae836ec2b4d3f453023b9853d9d07c
bop-md5key: 0xeb3ad6ba929441a0ff64083d021c07f1
ors40595{}129:
```

Note. The bop-shakey and bop-md5key values must be recomputed and copied to the server any time a user's password is changed.

LDAP Accounting Attributes

Logging and accounting features include Account Start, Stop and Fail Times, and Dynamic Log. Typically, the Login and Logout logs can be accessed from the directory server software. Additional third-party software is required to retrieve and reset the log information to the directory servers for billing purposes.

The following sections describe accounting server attributes.

AccountStartTime

User account start times are tracked in the AccountStartTime attribute of the user's directory entry that keeps the time stamp and accounting information of user log-ins. The following fields (separated by carriage returns "␣") are contained in the Login log. Some fields are only used for Layer 2 Authentication.

Fields Included For Any Type of Authentication

- User account ID or username client entered to log-in: variable length digits.
- Time Stamp (YYYYMMDDHHMMSS (YYYY:year, MM:month, DD:day, HH:hour, MM:minute, SS:second))
- Switch serial number: Alcatel-Lucent.BOP.<switch name>.<MAC address>
- Client IP address: variable length digits.

Fields Included for Layer 2 Authentication Only

- Client MAC address: xx:xx:xx:xx:xx:xx:xx (alphanumeric).
- Switch VLAN number client joins in multiple authority mode (0=single authority; 2=multiple authority); variable-length digits.
- Switch slot number to which client connects: nn
- Switch port number to which client connects: nn
- Switch virtual interface to which client connects: nn

AccountStopTime

User account stop times are tracked in the AccountStopTime attribute that keeps the time stamp and accounting information of successful user log-outs. The same fields as above (separated by carriage returns “\n”) are contained in the Logout log. A different carriage return such as the # sign may be used in some situations. Additionally, these fields are included but apply only to the Logout log:

Fields For Any Type of Authentication

- Log-out reason code, for example LOGOFF(18) or DISCONNECTED BY ADMIN(19)
- User account ID or username client entered to log-in: variable length digits.

Fields For Layer 2 Authentication Only

- Number of bytes received on the port during the client’s session from log-in to log-out: variable length digits.
- Number of bytes sent on the port during the client’s session from log-in to log-out: variable length digits.
- Number of frames received on the port during the client’s session from log-in to log-out: variable length digits.
- Number of frames sent on the port during the clients session from log-in to log-out: variable length digits.

AccountFailTime

The AccountFailTime attribute log records the time stamp and accounting information of unsuccessful user log-ins. The same fields in the Login Log—which are also part of the Logout log (separated by carriage returns “\n”)—are contained in the Login Fail log. A different carriage return such as the # sign may be used in some situations. Additionally, these fields are included but apply only to the Login Fail log.

- User account ID or username client entered to log-in: variable length digits.
- Log-in fail error code: nn. For error code descriptions refer to the vendor-specific listing for the specific directory server in use.
- Log-out reason code, for example PASSWORD EXPIRED(7) or AUTHENTICATION FAILURE(21).

Dynamic Logging

Dynamic logging may be performed by an LDAP-enabled directory server if an LDAP server is configured **first** in the list of authentication servers configured through the **aaa accounting session** command. Any other servers configured are used for accounting (storing history records) only. For example:

```
-> aaa accounting session ldap2 rad1 rad2
```

In this example, server **ldap2** will be used for dynamic logging, and servers **rad1** and **rad2** will be used for accounting.

If you specify a RADIUS server first, all of the servers specified will be used for recording history records (not logging). For example:

```
-> aaa accounting session rad1 ldap2
```

In this example, both the **rad1** and **ldap2** servers will be used for history only. Dynamic logging will not take place on the LDAP server.

Dynamic entries are stored in the LDAP-enabled directory server database from the time the user successfully logs in until the user logs out. The entries are removed when the user logs out.

- Entries are associated with the switch the user is logged into.
- Each dynamic entry contains information about the user's connection. The related attribute in the server is bop-loggedusers.

A specific object class called **alcatelBopSwitchLogging** contains three attributes as follows:

Attribute	Description
bop-basemac	MAC range, which uniquely identifies the switch.
bop-switchname	Host name of the switch.
bop-loggedusers	Current activity records for every user logged onto the switch identified by bop-basemac.

Each switch that is connected to the LDAP-enabled directory server will have a DN starting with bop-basemac-xxxxx, ou=bop-logging. If the organizational unit ou=bop.logging exists somewhere in the tree under searchbase, logging records are written on the server. See the server manufacturer's documentation for more information about setting up the server.

The `bop-loggedusers` attribute is a formatted string with the following syntax:

loggingMode : accessType ipAddress port macAddress vlanList userName

The fields are defined here:

Field	Possible Values
loggingMode	ASA <i>x</i> —for an authenticated user session, where <i>x</i> is the number of the session
accessType	Any one of the following: CONSOLE, MODEM, TELNET, HTTP, FTP, XCAP
ipAddress	The string IP followed by the IP address of the user.
port	The string PORT followed by the slot/port number.
macAddress	The string MAC followed by the MAC address of the user.
vlanList	The string VLAN followed by the list of VLANs the user is authorized (for single-mode authority).
userName	The login name of the user.

For example:

```
"ASA      0      : CONSOLE IP 65.97.233.108   Jones"
```

Configuring the LDAP Authentication Client

Use the [aaa tacacs+-server](#) command to configure LDAP authentication parameters on the switch. The server name, host name or IP address, distinguished name, password, and the search base name are required for setting up the server. Optionally, a backup host name or IP address may be configured, as well as the number of retransmit tries, the timeout for authentication requests, and whether or not a secure Socket Layer (SSL) is enabled between the switch and the server.

Note. The server should be configured with the appropriate schema before the `aaa ldap-server` command is configured.

The keywords for the `aaa ldap-server` command are listed here:

Required for creating:	optional:
host	type
dn	retransmit
password	timeout
base	port
	ssl

Creating an LDAP Authentication Server

An example of creating an LDAP server:

```
-> aaa ldap-server ldap2 host 10.10.3.4 dn cn=manager password tpub base c=us
```

In this example, the switch will be able to communicate with an LDAP server (called **ldap2**) that has an IP address of 10.10.3.4, a domain name of cn=manager, a password of tpub, and a searchbase of c=us. These parameters must match the same parameters configured on the server itself.

Note. The distinguished name must be different from the searchbase name.

Modifying an LDAP Authentication Server

To modify an LDAP authentication server, use the **aaa ldap-server** command with the server name; or, if you have just entered the **aaa ldap-server** command to create or modify the server, you can use command prefix recognition. For example:

```
-> aaa ldap-server ldap2 password my_pass
-> timeout 4
```

In this example, an existing LDAP server is modified with a different password, and then the timeout is modified on a separate line. These two command lines are equivalent to:

```
-> aaa ldap-server ldap2 password my_pass timeout 4
```

Setting Up SSL for an LDAP Authentication Server

A Secure Socket Layer (SSL) may be set up on the server for additional security. When SSL is enabled, the server's identity will be authenticated. The authentication requires a certificate from a Certification Authority (CA). If the CA providing the certificate is well-known, the certificate is automatically extracted from the **Kbase.img** file on the switch (**certs.pem**). If the CA is not well-known, the CA's certificate must be transferred to the switch via FTP to the /flash/certified or /flash/working directory and should be named **optcerts.pem**. The switch merges either or both of these files into a file called **ldapcerts.pem**.

To set up SSL on the server, specify **ssl** with the **aaa ldap-server** command:

```
-> aaa ldap-server ldap2 ssl
```

The switch automatically sets the port number to 636 when SSL is enabled. The 636 port number is typically used on LDAP servers for SSL. The port number on the switch must match the port number configured on the server. If the port number on the server is different from the default, use the **aaa ldap-server** command with the **port** keyword to configure the port number. For example, if the server port number is 635, enter the following:

```
-> aaa ldap-server ldap2 port 635
```

The switch will now be able to communicate with the server on port 635.

To remove SSL from the server, use **no** with the **ssl** keyword. For example:

```
-> aaa ldap-server ldap2 no ssl
```

SSL is now disabled for the server.

Removing an LDAP Authentication Server

To delete an LDAP server from the switch configuration, use the **no** form of the command with the relevant server name.

```
-> no aaa ldap-server topanga5
```

The topanga5 server is removed from the configuration.

Verifying the Authentication Server Configuration

To display information about authentication servers, use the following command:

show aaa server Displays information about a particular AAA server or AAA servers.

An example of the output for this command is given in [“Quick Steps For Configuring Authentication Servers” on page 31-4](#). For more information about the output of this command, see the *OmniSwitch 6250 CLI Reference Guide*.

32 Configuring 802.1X

Physical devices attached to a LAN port on the switch through a point-to-point LAN connection may be authenticated through the switch through port-based network access control. This control is available through the IEEE 802.1X standard implemented on the switch. .

In This Chapter

This chapter describes 802.1X ports used for port-based access control and how to configure them through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

This chapter provides an overview of 802.1X and includes the following information:

- [“Setting Up Port-Based Network Access Control” on page 32-9](#)
- [“Enabling 802.1X on Ports” on page 32-9](#)
- [“Setting 802.1X Switch Parameters” on page 32-9](#)
- [“Configuring 802.1X Port Parameters” on page 32-10](#)
- [“Verifying the 802.1X Port Configuration” on page 32-13](#)

802.1X Specifications

RFCs Supported	RFC 2284–PPP Extensible Authentication Protocol (EAP) RFC 2865–Remote Authentication Dial In User Service (RADIUS) RFC 2866–RADIUS Accounting RFC 2867–RADIUS Accounting Modifications for Tunnel Protocol Support RFC 2868–RADIUS Attributes for Tunnel Protocol Support RFC 2869–RADIUS Extensions
IEEE Standards Supported	IEEE 802.1X-2001–Standard for Port-based Network Access Control 802.1X RADIUS Usage Guidelines
Platforms Supported	OmniSwitch 6250 Series
Maximum number of 802.1x users per NI module.	256 (supplicants and non-supplicants)

802.1X Defaults

The following table lists the defaults for 802.1X port configuration through the **802.1x** command and the relevant command keywords:

Description	Keyword	Default
Port control in both directions or incoming only.	direction {both in}	both
Port control authorized on the port.	port control {force-authorized force-unauthorized auto}	auto
The time during which the port will not accept an 802.1X authentication attempt.	quiet-period	60 seconds
The time before an EAP Request Identity will be re-transmitted.	tx-period	30 seconds
Number of seconds before the switch will time out an 802.1X user who is attempting to authenticate.	supp-timeout	30 seconds
Number of times to poll a device for EAP frames to determine whether or not the device is an 802.1x client.	supp-polling retry	2
Maximum number of times the switch will retransmit an authentication request before it times out.	max-req	2
Amount of time that must expire before a re-authentication attempt is made.	re-authperiod	3600 seconds
Whether or not the port is re-authenticated.	no reauthentication reauthentication	no reauthentication

Note. By default, accounting is disabled for 802.1X authentication sessions.

Quick Steps for Configuring 802.1X

1 Configure the port as a mobile port and then as an 802.1X port using the following **vlan port** commands:

```
-> vlan port mobile 3/1
-> vlan port 3/1 802.1x enable
```

The port is set up automatically with 802.1X defaults. See “[802.1X Defaults](#)” on page 32-2 for information about the defaults. For more information about **vlan port** commands, see [Chapter 6, “Assigning Ports to VLANs.”](#)

2 Configure the RADIUS server to be used for port authentication:

```
-> aaa radius-server rad1 host 10.10.2.1 timeout 25
```

See [Chapter 31, “Managing Authentication Servers,”](#) for more information about configuring RADIUS authentication servers for 802.1X authentication.

3 Associate the RADIUS server (or servers) with authentication for 802.1X ports:

```
-> aaa authentication 802.1x rad1
```

4 (Optional) Associate the server (or servers) to be used for accounting (logging) 802.1X sessions. For example:

```
-> aaa accounting 802.1x rad2 ldap3 local
```

5 (Optional) Configure port-access control parameters for the 802.1X port using the **802.1x** command:

```
-> 802.1x 3/1 quiet-period 45 max-req 3
```

6 (Optional) Configure the number of times supplicant devices are polled for identification using the **802.1x supp-polling retry** command:

```
-> 802.1x 3/1 supp-polling retry 10
```

Note. Verify the 802.1X port configuration using the **802.1x** command:

```
-> show 802.1x 1/13
802.1x configuration for slot 1 port 13:

direction                = both,
operational directions   = both,
port-control              = auto,
quiet-period (seconds)   = 60,
tx-period (seconds)      = 30,
supp-timeout (seconds)   = 30,
server-timeout (seconds) = 30,
max-req                   = 2,
re-authperiod (seconds)  = 3600,
reauthentication         = no
Supplicant polling retry count = 2
```

Optional. To display the number of 802.1x users on the switch, use the **show 802.1x users** command:

```
-> show 802.1x users
```

Slot	MAC	Port	User
------	-----	------	------

Port	Address	State	Name
3/1	00:60:4f:11:22:33	Connecting	user50
3/1	00:60:4f:44:55:66	Held	user51
3/1	00:60:4f:77:88:99	Authenticated	user52
3/3	00:60:22:15:22:33	Force-authenticated	N/A
3/3	00:60:22:44:75:66	Force-authenticated	N/A
3/3	00:60:22:37:98:09	Force-authenticated	N/A

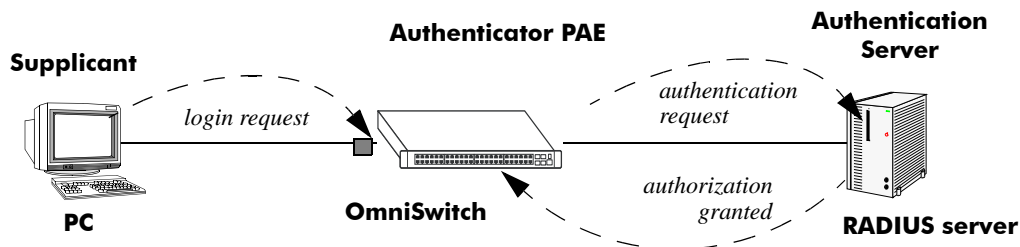
See the *OmniSwitch 6250 CLI Reference Guide* for information about the fields in this display.

802.1X Overview

The 802.1X standard defines port-based network access controls, and provides the structure for authenticating physical devices attached to a LAN. It uses the Extensible Authentication Protocol (EAP).

There are three components for 802.1X:

- **The Supplicant**—This is the device connected to the switch that supports the 802.1x protocol. The device may be connected directly to the switch or via a point-to-point LAN segment. Typically the supplicant is a PC or laptop.
- **The Authenticator Port Access Entity (PAE)**—This entity requires authentication from the supplicant. The authenticator is connected to the supplicant directly or via a point-to-point LAN segment. The OmniSwitch acts as the authenticator.
- **The Authentication Server**—This component provides the authentication service and verifies the credentials (username, password, challenge, etc.) of the supplicant. On the OmniSwitch, only RADIUS servers are currently supported for 802.1X authentication.



802.1X Components

Note. The OmniSwitch itself cannot be an 802.1X supplicant.

A device that does not use the 802.1x protocol for authentication is referred to as a *non-supplicant*. The Access Guardian feature provides configurable device classification policies to authenticate access of both supplicant and non-supplicant devices on 802.1x ports. See [Chapter 30, “Configuring Access Guardian,”](#) for more information.

Supplicant Classification

When an EAP frame or an unknown source data frame is received from a supplicant, the switch sends an EAP packet to request the supplicant’s identity. The supplicant then sends the information (an EAP response), which is validated on an authentication server set up for authenticating 802.1X ports. The server determines whether additional information (a challenge, or secret) is required from the supplicant.

After the supplicant is successfully authenticated, the MAC address of the supplicant is learned in the appropriate VLAN depending on the following conditions:

- If the authentication server returned a VLAN ID, then the supplicant is assigned to that VLAN. All subsequent traffic from the supplicant is then forwarded on that VLAN.

- If the authentication server does not return a VLAN ID, then the supplicant is classified according to any device classification policies that are configured for the port. See [Chapter 30, “Configuring Access Guardian,”](#) for more information.
- If the authentication server does not return a VLAN ID and there are no user-configured device classification policies for the port, then by default Group Mobility is used to classify the supplicant. If Group Mobility is unable to classify the supplicant, then the supplicant is assigned to the default VLAN for the 802.1X port.
- If the authentication server returns a VLAN ID that does not exist or authentication fails, the supplicant is blocked.

Note that multiple supplicants can be authenticated on a given 802.1X port. Each supplicant MAC address received on the port is authenticated and learned separately. Only those that authenticate successfully are allowed on the port, as described above. Those that fail authentication are blocked on the 802.1X port.

The global configuration of this feature is controlled by the **aaa authentication 802.1x** command. This command enables 802.1X for the switch and identifies the primary and backup authentication servers. See [“Setting 802.1X Switch Parameters” on page 32-9](#) for more information about configuring this command.

Using the **802.1x** command, an administrator may force an 802.1X port to always accept any frames on the port (therefore not requiring a device to first authenticate on the port); or an administrator may force the port to never accept any frames on the port. See [“Configuring the Port Authorization” on page 32-10](#).

802.1X Ports and DHCP

DHCP requests on an 802.1X port are treated as any other traffic on the 802.1X port.

When the port is in an unauthorized state (which means no device has authenticated on the port), the port is blocked from receiving any traffic except 802.1X packets. This means that DHCP requests will be blocked as well.

When the port is in a forced unauthorized state (the port is manually set to unauthorized), the port is blocked from receiving all traffic, including 802.1X packets and DHCP requests.

If the port is in a forced authorized state (manually set to authorized), any traffic, including DHCP, is allowed on the port.

If the port is in an authorized state because a device has authenticated on the port, only traffic with an authenticated MAC address is allowed on the port. DHCP requests from the authenticated MAC address are allowed; any others are blocked.

Re-authentication

After a supplicant has successfully authenticated through an 802.1X port, the switch may be configured to periodically re-authenticate the supplicant (re-authentication is disabled by default). In addition, the supplicant may be manually re-authenticated (see [“Re-authenticating an 802.1X Port” on page 32-11](#)).

The re-authentication process is transparent to a user connected to the authorized port. The process is used for security and allows the authenticator (the OmniSwitch) to maintain the 802.1X connection.

Note. If the MAC address of the supplicant has aged out during the authentication session, the 802.1X software in the switch will alert the source learning software in the switch to re-learn the address.

802.1X ports may also be initialized if there a problem on the port. Initializing a port drops connectivity to the port and requires the port to be re-authenticated. See [“Initializing an 802.1X Port” on page 32-12](#).

802.1X Accounting

802.1X authentication sessions may be logged if servers are set up for 802.1X accounting. Accounting may also be done through the local Switch Logging feature. For information about setting up accounting for 802.1X, see [“Configuring Accounting for 802.1X” on page 32-12](#).

Setting Up Port-Based Network Access Control

For port-based network access control, 802.1X must be enabled for the switch and the switch must know which servers to use for authenticating 802.1X supplicants.

In addition, 802.1X must be enabled on each port that is connected to an 802.1X supplicant (or device). Optional parameters may be set for each 802.1X port.

The following sections describe these procedures in detail.

Setting 802.1X Switch Parameters

Use the **aaa authentication 802.1x** command to enable 802.1X for the switch and specify an authentication server (or servers) to be used for authenticating 802.1X ports. The servers must already be configured through the **aaa radius-server** command. An example of specifying authentication servers for authenticating all 802.1X ports on the switch:

```
-> aaa authentication 802.1x rad1 rad2
```

In this example, the **rad1** server will be used for authenticating 802.1X ports. If **rad1** becomes unavailable, the switch will use **rad2** for 802.1X authentication. When this command is used, 802.1X is automatically enabled for the switch.

Enabling MAC Authentication

Use the **aaa authentication mac** command to enable MAC authentication for the switch and specify an authentication server (or servers) to be used for authenticating non-supplicants on 802.1x ports. As with enabling 802.1x authentication, the servers specified with this command must already be configured through the **aaa radius-server** command.

The following example command specifies authentication servers for authenticating non-supplicant devices on 802.1x ports:

```
-> aaa authentication mac rad1 rad2
```

Note that the same RADIUS servers can be used for 802.1x (supplicant) and MAC (non-supplicant) authentication. Using different servers for each type of authentication is allowed but not required.

For more information about using MAC authentication and classifying non-supplicant devices, see [Chapter 30, “Configuring Access Guardian.”](#)

Enabling 802.1X on Ports

To enable 802.1X on a port, use the **vlan port 802.1x** command. The port must first be configured as a mobile port.

```
-> vlan port mobile 3/1  
-> vlan port 3/1 802.1x enable
```

The **vlan port 802.1x** command enables 802.1X on port 1 of slot 3. The port will be set up with defaults listed in [“802.1X Defaults” on page 32-2.](#)

To disable 802.1X on a port, use the **disable** option with **vlan port 802.1x** command. For more information about **vlan port** commands, See [Chapter 6, “Assigning Ports to VLANs.”](#)

Configuring 802.1X Port Parameters

By default, when 802.1X is enabled on a port, the port is configured for bidirectional control, automatic authorization, and re-authentication. In addition, there are several timeout values that are set by default as well as a maximum number of times the switch will retransmit an authentication request to the user.

All of these parameters may be configured on the same command line but are shown here configured separately for simplicity.

Configuring the Port Control Direction

To configure the port control direction, use the **802.1x** command with the **direction** keyword with **both** for bidirectional or **in** for incoming traffic only. For example:

```
-> 802.1x 3/1 direction in
```

In this example, the port control direction is set to incoming traffic only on port 1 of slot 3.

The type of port control (or authorization) is configured with the **port-control** parameter described in the next section.

Configuring the Port Authorization

Port authorization determines whether the port is open to all traffic, closed to all traffic, or open to traffic after the port is authenticated. To configure the port authorization, use the **802.1x** command with the **port-control** keyword and the **force-authorized**, **force-unauthorized**, or **auto** option.

```
-> 802.1x 3/1 port-control force-authorized
```

In this example, the port control on port 1 of slot 3 is always authorized for any traffic.

The **auto** option configures the port to be open for traffic when a device successfully completes an 802.1X authentication exchange with the switch.

Configuring 802.1X Port Timeouts

There are several timeouts that may be modified per port:

- Quiet timeout—The time during which the port will not accept an 802.1X authentication attempt after an authentication failure.
- Transmit timeout—The time before an EAP Request Identity message will be re-transmitted.
- Supplicant (or user) timeout—The time before the switch will timeout an 802.1X user who is attempting to authenticate. During the authentication attempt, the switch sends requests for authentication information (identity requests, challenge response, etc.) to the supplicant (see [“Configuring the Maximum Number of Requests” on page 32-11](#)). If the supplicant does not reply to these requests, the supplicant is timed out when the timeout expires.

To modify the quiet timeout, use the **802.1x** command with the **quiet-period** keyword. To modify the transmit timeout, use the **802.1x** command with the **tx-period** keyword. To modify the supplicant or user timeout, use the **802.1x** command with the **supp-timeout** keyword. For example:

```
-> 802.1x 3/1 quiet-period 50 tx-period 25 supp-timeout 25
```

This command changes the quiet timeout to 50 seconds; the transmit timeout to 25 seconds; and the user timeout to 25 seconds.

Note. The authentication server timeout may also be configured (with the **server-timeout** keyword) but the value is always superseded by the value set for the RADIUS server through the **aaa radius-server** command.

Configuring the Maximum Number of Requests

During the authentication process, the switch sends requests for authentication information from the supplicant. By default, the switch will send up to two requests for information. If the supplicant does not reply within the timeout value configured for the supplicant timeout, the authentication session attempt will expire. The switch will then use its quiet timeout and transmit timeout before accepting an authentication attempt or sending out an identity request.

To change the maximum number of requests sent to the supplicant during an authentication attempt, use the **max-req** keyword with the **802.1x** command. For example:

```
-> 802.1x 3/1 max-req 3
```

In this example, the maximum number of requests that will be sent is three.

Configuring the Number of Polling Retries

To change the number of times a device is polled for EAP frames to determine whether or not the device is an 802.1x client, use the **802.1x supp-polling retry** command. For example:

```
-> 802.1x 3/1 supp-polling retry 10
```

In this example, the maximum number of times a device is polled is set to 10. If no EAP frames are received, the device is considered a non-suppliant, and any non-suppliant classification policies configured for the port are applied to the device.

To bypass 802.1x authentication and classify supplicants connected to the port as non-suplicants, set the number of polling retries to zero:

```
-> 802.1x 3/1 supp-polling retry 0
```

Note. Setting the number of polling retries to zero turns off 802.1x authentication for the port; all devices (including supplicants) are then classified as non-suplicants. As a result, non-suppliant policies that use MAC-based authentication are now applicable to supplicant devices, not just non-suppliant devices.

Re-authenticating an 802.1X Port

An automatic reauthentication process may be enabled or disabled on any 802.1X port. The re-authentication is used to maintain the 802.1X connection (not to re-authenticate the user). The process is transparent to the 802.1X supplicant. By default, re-authentication is not enabled on the port.

To enable or disable re-authentication, use the **reauthentication** or **no reauthentication** keywords with the **802.1x** command. For example:

```
-> 802.1x 3/1 reauthentication
```

In this example, re-authentication will periodically take place on port 1 of slot 3.

The **re-authperiod** parameter may be used to configure the time that must expire before automatic re-authentication attempts. For example:

```
-> 802.1x 3/1 reauthentication re-authperiod 25
```

In this example, automatic re-authentication is enabled, and re-authentication will take place on the port every 25 seconds.

To manually re-authenticate a port, use the **802.1x re-authenticate** command. For example:

```
-> 802.1x re-authentication 3/1
```

This command initiates a re-authentication process for port 1 on slot 3.

Initializing an 802.1X Port

An 802.1X port may be reinitialized. This is useful if there is a problem on the port. The reinitialization process drops connectivity with the supplicant and forces the supplicant to be re-authenticated. Connectivity is restored with successful re-authentication. To force an initialization, use the **802.1x initialize** command with the relevant slot/port number. For example:

```
-> 802.1x initialize 3/1
```

This command drops connectivity on port 1 of slot 3. The switch sends out a Request Identity message and restores connectivity when the port is successfully re-authenticated.

Configuring Accounting for 802.1X

To log 802.1X sessions, use the **aaa accounting 802.1x** command with the desired RADIUS server names; use the keyword **local** to specify that the Switch Logging function in the switch should be used to log 802.1X sessions. RADIUS servers are configured with the **aaa radius-server** command.

```
-> aaa accounting 802.1x rad1 local
```

In this example, the RADIUS server **rad1** will be used for accounting. If **rad1** becomes unavailable, the local Switch Logging function in the switch will log 802.1X sessions. For more information about Switch Logging, see [Chapter 39, "Using Switch Logging."](#)

Verifying the 802.1X Port Configuration

A summary of the **show** commands used for verifying the 802.1X port configuration is given here:

show 802.1x users	Displays a list of all users (supplicants) for one or more 802.1X ports.
show 802.1x non-supplicant	Displays a list of all non-802.1x users (non-supplicants) learned on one or more 802.1x ports.
show 802.1x statistics	Displays statistics about 802.1X ports.
show 802.1x device classification policies	Displays Access Guardian 802.1x device classification policies configured for 802.1x ports.
show aaa authentication 802.1x	Displays information about the global 802.1X configuration on the switch.
show aaa accounting 802.1x	Displays information about accounting servers configured for 802.1X port-based network access control.
show aaa authentication mac	Displays a list of RADIUS servers configured for MAC based authentication.

For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

33 Managing Policy Servers

Quality of Service (QoS) policies that are configured through Alcatel-Lucent's PolicyView network management application are stored on a Lightweight Directory Access Protocol (LDAP) server. PolicyView is an OmniVista application that runs on an attached workstation.

In This Chapter

This chapter describes how LDAP directory servers are used with the switch for policy management. There is no required configuration on the switch. When policies are created on the directory server through PolicyView, the PolicyView application automatically configures the switch to communicate with the server. This chapter includes information about modifying configuration parameters through the Command Line Interface (CLI) if manual reconfiguration is necessary. For more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Throughout this chapter the term *policy server* is used to refer to LDAP directory servers used to store policies. Procedures described in this chapter include:

- [“Installing the LDAP Policy Server” on page 33-3](#)
- [“Modifying Policy Servers” on page 33-4](#)
- [“Verifying the Policy Server Configuration” on page 33-7](#)

Policy Server Specifications

The following table lists important information about LDAP policy servers:

LDAP Policy Servers RFCs Supported	RFC 2251–Lightweight Directory Access Protocol (v3) RFC 3060–Policy Core Information Model—Version 1 Specification
Platforms Supported	OmniSwitch 6250 Series
Maximum number of policy servers (supported on the switch)	4
Maximum number of policy servers (supported by PolicyView)	1

Policy Server Defaults

Defaults for the **policy server** command are as follows:

Description	Keyword	Default
The port number for the server	port	389 (SSL disabled) 636 (SSL enabled)
Priority value assigned to a server, used to determine search order	preference	0 (lowest)
Whether a Secure Socket Layer is configured for the server	ssl no ssl	no ssl

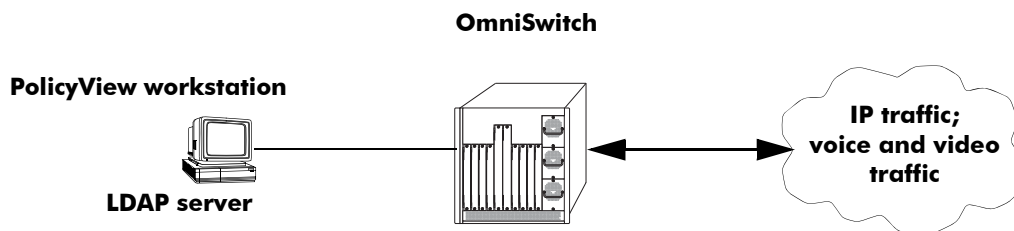
Policy Server Overview

The Lightweight Directory Access Protocol (LDAP) is a standard directory server protocol. The LDAP policy server client in the switch is based on RFC 2251. Currently, only LDAP servers are supported for policy management.

When the policy server is connected to the switch, the switch is automatically configured to communicate with the server to download and manage policies created by the PolicyView application. There is no required user configuration. (Note that the LDAP policy server is automatically installed when the PolicyView application is installed.)

Note. The switch has separate mechanisms for managing QoS policies stored on an LDAP server and QoS policies configured directly on the switch. For more information about creating policies directly on the switch, see [Chapter 34, “Configuring QoS.”](#)

Information about installing the LDAP policy server is included in this chapter. Consult the server manufacturer’s documentation for detailed information about configuring the server.



Policy Server Setup

Installing the LDAP Policy Server

Currently Netscape Directory Server 4.15 is supported. The server software is bundled with the PolicyView NMS application.

- 1 Install the directory server software on the server.
- 2 Install the Java Runtime Environment on the server.

See your server documentation for additional details on setting up the server.

See the next sections of this chapter for information about modifying policy server parameters or viewing information about policy servers.

Modifying Policy Servers

Policy servers are automatically configured when the server is installed; however, policy server parameters may be modified if necessary.

Note. SSL configuration must be done manually through the **policy server** command.

Modifying LDAP Policy Server Parameters

Use the **policy server** command to modify parameters for an LDAP policy server.

Keywords for the command are listed here:

Policy server keywords

port	password
admin	searchbase
preference	ssl
user	

For information about policy server parameter defaults, see [“Policy Server Defaults” on page 33-2](#).

Disabling the Policy Server From Downloading Policies

Policy servers may be prevented from downloading policies to the switch. By default, policy servers are enabled to download policies.

To disable a server, use the **policy server** command with the **admin** keyword and **down** option.

```
-> policy server 10.10.2.3 admin down
```

In this example, an LDAP server with an IP address of 10.10.2.3 will not be used to download policies. Any policies already downloaded to the switch are not affected by disabling the server.

To re-enable the server, specify **up**.

```
-> policy server 10.10.2.3 admin up
```

The server is now available for downloading policies.

To delete a policy server from the configuration, use the **no** form of the command with the relevant IP address:

```
-> no policy server 10.10.2.3
```

If the policy server is not created on the default port, the **no** form of the command must include the port number. For example:

```
-> no policy server 10.10.2.4 5000
```

Modifying the Port Number

To modify the port, enter the **policy server** command with the **port** keyword and the relevant port number.

```
-> policy server 10.10.2.3 port 5000
```

Note that the port number must match the port number configured on the policy server.

If the port number is modified, any existing entry for that policy server is not removed. Another entry is simply added to the policy server table.

Note. If you enable SSL, the port number is automatically set to 636. (This does not create another entry in the port table.)

For example, if you configure a policy server with port 389 (the default), and then configure another policy server with the same IP address but port number 5000, two entries will display on the **show policy server** screen.

```
-> policy server 10.10.2.3
-> policy server 10.10.2.3 port number 5000
-> show policy server
```

Server	IP Address	port	enabled	status	primary
1	10.10.2.3	389	Yes	Up	X
2	10.10.2.3	5000	No	Down	-

To remove an entry, use the **no** form of the **policy server** command. For example:

```
-> no policy server 10.10.2.3 port number 389
```

The first entry is removed from the policy server table.

Modifying the Policy Server Username and Password

A user name and password may be specified so that only specific users can access the policy server.

```
-> policy server 10.10.2.3 user kandinsky password blue
```

If this command is entered, a user with a username of **kandinsky** and a password of **blue** will be able to access the LDAP server to modify parameters on the server itself.

Modifying the Searchbase

The searchbase name is "o=alcatel.com" by default. To modify the searchbase name, enter the **policy server** command with the **searchbase** keyword. For example:

```
-> policy server 10.10.2.3 searchbase "ou=qo,o=company,c=us"
```

Note that the searchbase path must be a valid path in the server directory structure.

Configuring a Secure Socket Layer for a Policy Server

A Secure Socket Layer (SSL) may be configured between the policy server and the switch. If SSL is enabled, the PolicyView application can no longer write policies to the LDAP directory server.

By default, SSL is disabled. To enable SSL, use the **policy server** command with the **ssl** option. For example:

```
-> policy server 10.10.2.3 ssl
```

SSL is now enabled between the specified server and the switch. The port number in the switch configuration will be automatically set to 636, which is the port number typically used for SSL; however, the port number should be configured with whatever port number is set on the server. For information about configuring the port number, see [“Modifying the Port Number” on page 33-5](#).

To disable SSL, use **no ssl** with the command:

```
-> policy server 10.10.2.3 no ssl
```

SSL is disabled for the 10.10.2.3 policy server. No additional policies may be saved to the directory server from the PolicyView application.

Loading Policies From an LDAP Server

To download policies (or rules) from an LDAP server to the switch, use the **policy server load** command. Before a server can download policies, it must also be set up and operational (able to bind).

To download policies from the server, enter the following:

```
-> policy server load
```

Use the **show policy server long** command to display the last load time. For example:

```
-> show policy server long
LDAP server 0
  IP address           : 10.10.2.3,
  TCP port             : 16652,
  Enabled              : Yes,
  Operational Status   : Down,
  Preference           : 99,
  Authentication       : password,
  SSL                  : Disabled,
  login DN             : cn=DirMgr
  searchbase           : o=company
  Last load time       : 02/14/02 16:38:18
```

Removing LDAP Policies From the Switch

To flush LDAP policies from the switch, use the **policy server flush** command. Note that any policies configured directly on the switch through the CLI *are not affected* by this command.

```
-> policy server flush
```

Interaction With CLI Policies

Policies configured via PolicyView can only be modified through PolicyView. They cannot be modified through the CLI. Any policy management done through the CLI only affects policies configured through the CLI. For example, the **qos flush** command only removes CLI policies; LDAP policies are not affected.

Also, the **policy server flush** command removes only LDAP policies; CLI policies are not affected.

Note. If policies are applied from PolicyView or vice versa, it will activate all current configuration.

For more information about configuring policies through the CLI, see [Chapter 34, “Configuring QoS.”](#)

Verifying the Policy Server Configuration

To display information about authentication and policy servers, use the following commands:

show policy server	Displays information about servers from which policies may be downloaded to the switch.
show policy server long	Displays detailed information about an LDAP policy server.
show policy server statistics	Displays statistics about policy directory servers.
show policy server rules	Displays the names of policies originating on a directory server that have been downloaded to the switch.
show policy server events	Displays any events related to a directory server.

34 Configuring QoS

Alcatel-Lucent's QoS software provides a way to manipulate flows coming through the switch based on user-configured policies. The flow manipulation (generally referred to as *Quality of Service* or *QoS*) may be as simple as allowing/denying traffic, or as complicated as remapping 802.1p bits from a Layer 2 network to ToS values in a Layer 3 network.

While policies may be used in many different types of network scenarios, there are several typical types discussed in this chapter:

- **Basic QoS**—includes traffic prioritization and bandwidth shaping.
- **ICMP policies**—includes filtering, prioritizing, and/or rate limiting ICMP traffic for security.
- **802.1p/ToS/DSCP**—includes policies for marking and mapping.
- **Policy Based Routing (PBR)**—includes policies for redirecting routed traffic.
- **Policy Based Mirroring**—includes mirror-to-port (MTP) policies for mirroring ingress, egress, or both ingress and egress traffic.
- **Access Control Lists (ACLs)**—ACLs are a specific type of QoS policy used for Layer 2 and Layer 3/4 filtering. Since filtering is used in many different network situations, ACLs are described in a separate chapter (see [Chapter 35, “Configuring ACLs”](#)).

In This Chapter

This chapter describes QoS in general and how policies are used on the switch. It provides information about configuring QoS through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Setting up global QoS parameters (see [page 34-16](#))
- Configuring QoS Ports and Queueing Schemes [page 34-23](#)
- Setting up policy components, such as policy conditions and actions (see [page 34-31](#))
- Configuring specific types of policies (see [page 34-62](#))

Note. Policies may also be configured through the PolicyView NMS application and stored on an attached LDAP server. LDAP policies are downloaded to the switch and managed via the Policy Manager feature in the switch. For more information about managing LDAP policies, see [Chapter 33, “Managing Policy Servers.”](#)

QoS Specifications

The QoS functionality described in this chapter is supported on the OmniSwitch 6250 Series switches, unless otherwise stated in the following QoS Specifications table or specifically noted within any other section of this chapter. Note that any maximum limits provided in the Specifications table are subject to available system resources.

Maximum number of policy rules	1280 (ingress and egress rules combined)
Maximum number of egress policy rules	512 (egress rules supported only on OmniSwitch 6250-Metro Models, Release 6.6.2.)
Maximum number of policy conditions	2048
Maximum number of policy actions	2048
Maximum number of policy validity periods	128
Maximum number of policy services	512
Maximum number of TCP and UDP port ranges	4
Maximum number of groups	1024 (VLAN groups supported only on OmniSwitch 6250-Metro Models, Release 6.6.2.)
Maximum number of group entries	1024 per group (512 per service group)
Maximum number of port groups per policy	8
Maximum number of rules per slot	1280
Maximum number of bandwidth shaping rules per slot	640
Maximum number of ToS or DSCP rules per slot	57
Maximum number of QoS policy lists per switch	13 (includes the default list)
Maximum number of priority queues per port	8
CLI Command Prefix Recognition	Some QoS commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

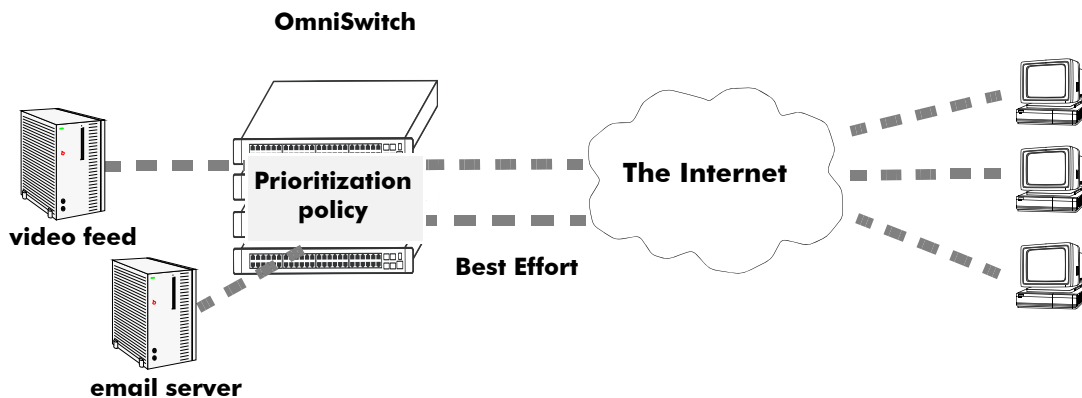
QoS General Overview

Quality of Service (QoS) refers to transmission quality and available service that is measured and sometimes guaranteed in advance for a particular type of traffic in a network. QoS lends itself to circuit-switched networks like ATM, which bundle traffic into cells of the same length and transmit the traffic over predefined virtual paths. In contrast, IP and other packet-switched networks operate on the concept of shared resources and *best effort* routing, using bandwidth as needed and reassembling packets at their destinations. Applying QoS to packet-switched networks requires different mechanisms than those used in circuit-switched networks.

QoS is often defined as a way to manage bandwidth. Another way to handle different types of flows and increased bandwidth requirements is to add more bandwidth. But bandwidth can be expensive, particularly at the WAN connection. If LAN links that connect to the WAN are not given more bandwidth, bottlenecks may still occur. Also, adding enough bandwidth to compensate for peak load periods will mean that at times some bandwidth will be unused. In addition, adding bandwidth does not guarantee any kind of control over network resources.

Using QoS, a network administrator can gain more control over networks where different types of traffic (or flows) are in use or where network congestion is high. Preferential treatment may be given to individual flows as required. Voice over IP (VoIP) traffic or mission-critical data may be marked as priority traffic and/or given more bandwidth on the link. QoS can also prevent large flows, such as a video stream, from consuming all the link's bandwidth. Using QoS, a network administrator can decide which traffic needs preferential treatment, and which traffic can be adequately served with best effort.

QoS is implemented on the switch through the use of user-defined policies. The following simplified illustration shows how video traffic may receive priority over email traffic.



Sample QoS Setup

QoS Policy Overview

A policy (or a *policy rule*) is made up of a condition and an action. The condition specifies parameters that the switch will examine in incoming flows, such as destination address or Type of Service (ToS) bits. The action specifies what the switch will do with a flow that matches the condition; for example, it may queue the flow with a higher priority, or reset the ToS bits.

Policies may be created directly on the switch through the CLI or WebView. Or policies may be created on an external LDAP server via the PolicyView application. The switch makes a distinction between policies created on the switch and policies created on an LDAP server.

Note. Policies may be only be modified using the same source used to create them. Policies configured through PolicyView may only be edited through PolicyView. Policies created directly on the switch through the CLI or WebView may only be edited on the switch. Policies may be created through the CLI or WebView, however, to override policies created in PolicyView. And vice versa.

This chapter discusses policy configuration using the CLI. For information about using WebView to configure the switch, see the *OmniSwitch 6250 Switch Management Guide*. For information about configuring policies through PolicyView, see the PolicyView online help.

How Policies Are Used

When a flow comes into the switch, the QoS software in the switch checks to see if there are any policies with conditions that match the flow.

- ***If there are no policies that match the flow***, the flow is accepted or denied based on the global disposition set for the switch. By default, the disposition is **accept**. Use the **qos default bridged disposition** or **qos default multicast disposition** command to change the disposition. If the flow is accepted, it is placed in a default queue on the output port.
- ***If there is more than one policy that matches the flow***, the policy with the highest precedence is applied to the flow. For more information about policy precedence, see [“Rule Precedence” on page 34-37](#).
- ***Flows must also match all parameters configured in a policy condition***. A policy condition must have at least one classification parameter.

Once the flow is classified and matched to a policy, the switch enforces the policy by mapping each packet of the flow to the appropriate queue and scheduling it on the output port. There are a total of eight queues per port. Traffic is mapped to a queue based on policies, the ToS/802.1p value of the packet, and whether the port is trusted or untrusted. For more information about queues, see [“QoS Ports and Queues” on page 34-23](#).

Valid Policies

The switch does not allow you to create invalid condition or action combinations; if you enter an invalid combination, an error message will display.

A list of valid condition and condition or action combinations is given in [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#).

It is possible to configure a valid QoS rule that is active on the switch, however the switch is not able to enforce the rule because some other switch function (for example, routing) is disabled. See the condition and condition/action combinations tables for more information about valid combinations ([“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#)).

Policy Lists

By default, QoS policy rules are applied to traffic ingressing on QoS ports. The ingress traffic is then bridged or routed to a destination port where the frames are serviced by the egress port/queue scheduler. Once the frames are serviced, policy rules can be applied to the frames before they are transmitted on the egress port.

Policy rules are *not* automatically applied to egress traffic. To apply a rule to egress traffic, the rule must belong to a QoS egress policy list. A policy list consists of a group of policy rules that is identified by the list name. There are three types of lists available:

- **Default**—All rules are associated with a default policy list when the rules are created. This list is not configurable, but it is possible to direct QoS not to assign a rule to this list. Default policy list rules are applied to ingress traffic.
- **Egress**—When a list is configured as an egress policy list, all rules associated with that list are applied to traffic egressing on QoS destination ports. Egress rules (members of an egress policy list) do not support all available policy actions and conditions. See [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#) to determine which conditions and actions are supported.

For more information, see [“Creating Policy Lists” on page 34-38](#).

Interaction With Other Features

QoS policies may be an integral part of configuring other switch features, such as Link Aggregation. In addition, QoS settings may affect other features in the switch; or QoS settings may require that other switch features be configured in a particular way.

A summary of related features is given here:

- **Dynamic Link Aggregates**—Policies may be used to prioritize dynamic link aggregation groups. For details, see [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)
- **802.1Q**—Tagged ports are always trusted, regardless of QoS settings. For information about configuring ports with 802.1Q, see [Chapter 22, “Configuring 802.1Q.”](#)
- **Mobile Ports**—Mobile ports are always trusted, regardless of QoS settings. For information about setting up mobile ports, see [Chapter 6, “Assigning Ports to VLANs.”](#)
- **LDAP Policy Management**—Policies may also be configured through the PolicyView application and stored on an attached LDAP server. LDAP policies may only be modified through PolicyView. For

information about setting up a policy server and managing LDAP policies, see [Chapter 33, “Managing Policy Servers.”](#)

Ethernet Service (VLAN Stacking)

- VLAN Stacking ports are always trusted and default classification is set to 802.1p.
- QoS policy rules take precedence over the VLAN Stacking SAP profile configuration. As a result, it is possible to configure QoS policy rules to override VLAN Stacking SAP profile settings, such as bandwidth and priority.
- Egress policy lists and VLAN translation Service Access Point (SAP) configurations are mutually exclusive. The switch only allows whichever of these two features is configured first.

For information about VLAN Stacking see [Chapter 9, “Configuring VLAN Stacking.”](#)

Condition Combinations

The CLI prevents you from configuring invalid condition combinations that are never allowed; however, it does allow you to create combinations that are supported in some scenarios. For example, you might configure **source ip** and a **destination ip** for the same condition.

The following conditions are supported and may be combined with other conditions and/or actions. Note that certain conditions are not supported when the condition is associated with an egress rule (a rule that is a member of an egress policy list. See [“Creating Policy Lists” on page 34-38](#) for more information).

Supported Policy Conditions Table

	Ingress Rules	Egress Rules (Egress Policy List)
Layer 1	destination port destination port group source port source port group	destination port destination port group
Layer 2	source MAC source MAC group destination MAC destination MAC group 802.1p ethertype source VLAN source VLAN group destination VLAN (multicast rules only) destination VLAN group (multicast rules only)	source MAC source MAC group destination MAC destination MAC group 802.1p ethertype source VLAN source VLAN group
Layer 3	IP protocol source IP multicast IP destination IP source network group destination network group multicast network group ToS, DSCP ICMP type, ICMP code source IPv6 destination IPv6 IPv6 traffic IPv6 next header (NH), IPv6 flow label (FL)	IP protocol source IP multicast IP destination IP source network group destination network group multicast network group ToS, DSCP
Layer 4	source TCP/UDP port destination TCP/UDP port service, service group TCP flags (ECN and CWR are not supported)	source TCP/UDP port destination TCP/UDP port service, service group TCP flags (ECN and CWR are not supported)
IP Multicast (IGMP)	destination only	

Consider the following guidelines when configuring policy conditions:

- The 802.1p and source VLAN conditions are the only Layer 2 conditions allowed in combination with Layer 3 IPv6 conditions.
- In a given rule, ToS or DSCP may be specified for a condition with priority specified for the action.
- IP multicast (IGMP) conditions can only be combined with destination conditions: destination slot/port, destination VLAN, destination MAC address, and destination IP address.

- The IP multicast condition works in combination with Layer 1, Layer 2, and Layer 3 destination conditions only if these conditions specify the device that sends the IGMP report packet.
- IP multicast traffic (not IGMP) is treated as regular traffic; QoS functionality works the same way with this type of traffic.
- The Layer 1 destination port condition only applies to bridged traffic, not routed traffic.
- Individual items and their corresponding groups cannot be combined in the same condition. For example, a source IP address cannot be included in a condition with a source IP network group.
- Layer 2 and Layer 3 rules are always effected on bridged and routed traffic. As a result, combining source or destination TCP/UDP port and IP protocol in a condition is allowed.
- If a policy rule contains only Layer 2 conditions, then the rule is applied only to Layer 2 traffic. To apply a pure Layer 2 rule to Layer 3 traffic, add the **source ip any** keywords to a condition for that rule. To apply a pure Layer 2 rule to IPv6 traffic, add the **ipv6** keyword to a condition for that rule.
- Unless the **ipv6** keyword is used in a policy condition, Layer 4 conditions apply only to IPv4 traffic.
- Classification of fragmented packets is not supported.

Use the following “Policy Condition Combinations Table” together with the “[Supported Policy Conditions Table](#)” as a guide when configuring policy conditions:

Policy Condition Combinations Table

	Layer 1	Layer 2	Layer 3*	Layer 4*	IP Multicast (IGMP)
Layer 1	All	All	All	All	destination only
Layer 2	All	All	All	except ethertype	destination only
Layer 3*	All	All	All	All	destination only
Layer 4*	All	except ethertype	All	All	None
IP Multicast (IGMP)	destination only	destination only	destination only	None	N/A

*IP multicast traffic (not IGMP) is treated as regular traffic; QoS functionality works the same way with this type of traffic, with the exception that the destination port condition does not apply.

For more information about combining policy actions or policy actions with conditions, see “[Action Combinations](#)” on page 34-9 and “[Condition and Action Combinations](#)” on page 34-11.

For specific information about how to configure policy conditions and actions to create a policy rule, see “[Creating Policies](#)” on page 34-31.

Action Combinations

The CLI prevents you from configuring invalid action combinations that are never allowed; however, it does allow you to create combinations that are supported in some scenarios. For example, an action specifying maximum bandwidth may be combined with an action specifying priority.

The following actions are supported and may be combined with other actions. Note that certain actions are not supported when the action is associated with an egress rule (a rule that is a member of an egress policy list. See [“Creating Policy Lists” on page 34-38](#) for more information).

Supported Policy Actions Table

Policy Action	Ingress Rules	Egress Rules (Egress Policy List)
ACL (disposition accept, drop, deny)	Yes	Yes
Priority/CoS	Yes	No
802.1p ToS/DCSP Stamping and Mapping (only applies to the outer 802.1p value; cannot modify the inner value)	Yes	Yes
Maximum Bandwidth	Yes	Yes
Maximum Depth	Yes	Yes
Tri-Color Marking (TCM) Rate Limiting	Yes	No
Shared (schedules multiple flows on the same queue when multiple rules use the same action)	Yes	Yes
Port Redirection	Yes	No
Link Aggregate Redirection (not supported on the OmniSwitch 6400, 6850, and 6855)	Yes	No
No Cache (disables the logging of rule entries to the hardware cache)	Yes	No
Port Disable	Yes	No
Permanent Gateway IP (not supported on the OmniSwitch 6400, 6850, and 6855)	Yes	No
Mirror	Yes	No

Consider the following guidelines when configuring policy actions:

- An 802.1p or ToS/DSCP action always sets the packet priority. For 802.1p marking, the priority is set according to the marked 802.1p. For ToS marking, the priority is set according to the marked ToS. For DSCP marking, the priority is set according to the marked DSCP.
- When 802.1p and priority marking are both set, priority is set according to 802.1p
- A ToS action alters the packet IP TOS fields. The DSCP bits 3,4,5 are reset to 0. For example, a ToS 2 action on a packet carrying DSCP 5 will give a DSCP of 40.
- A forwarding database entry (FDB) is not created for traffic dropped as the result of a policy drop action.

Use the following “Policy Action Combinations Table” together with the [“Supported Policy Actions Table”](#) as a guide when creating policy actions.

Policy Action Combinations Table

	Drop	Priority	Stamp/ Map	Max BW	Redirect Port	Redirect Linkagg	Port Disable	Permanent Gateway IP	Mirror
Drop	N/A	No	No	No	No	No	No	No	Yes
Priority	No	N/A	Yes	Yes	Yes	Yes	No	Yes	Yes
Stamp/Map	No	Yes	N/A	Yes	Yes	Yes	No	Yes	Yes
Max BW	No	Yes	Yes	N/A	Yes	Yes	No	Yes	Yes
Redirect Port	No	Yes	Yes	Yes	N/A	No	No	Yes	Yes
Redirect Linkagg	No	Yes	Yes	Yes	No	N/A	No	Yes	Yes
Port Disable	No	No	No	No	No	No	N/A	No	No
Permanent Gateway IP	No	Yes	Yes	Yes	Yes	Yes	No	N/A	Yes
Mirroring	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	N/A

For more information about combining policy conditions or policy conditions and actions, see [“Condition Combinations”](#) on page 34-7 and [“Condition and Action Combinations”](#) on page 34-11.

For specific information about how to configure policy conditions and actions to create a policy rule, see [“Creating Policies”](#) on page 34-31.

Condition and Action Combinations

Conditions and actions are combined in policy rules. The CLI prevents you from configuring invalid condition/action combinations that are never allowed; however, the following table provides a quick reference for determining which condition/action combinations are *not* valid. Each row represents a policy condition or conditions combined with the policy action or actions in the same row.

Policy Condition/Action Combinations

Conditions	Actions	Supported When?
multicast IP address <i>or</i> network group	all actions	never, except with disposition action
multicast IPv6 address	all actions	never, except with disposition and mirror actions
destination VLAN	all actions	never, except with disposition action in a multicast rule (a rule that uses the “multicast” keyword and only applies to IGMP traffic)
destination slot/port or port group	all actions	bridging only

Note. Additional policy condition/action combination restrictions may apply depending on whether the policy rule will apply to ingress or egress traffic. See [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#) for more information.

QoS Defaults

The following tables list the defaults for global QoS parameters, individual port settings, policy rules, and default policy rules.

Global QoS Defaults

Use the **qos reset** command is to reset global values to their defaults.

Description	Command	Default
QoS enabled or disabled	qos	enabled
Global default queuing scheme for ports	qos default servicing mode	strict priority queuing
Whether ports are globally trusted or untrusted	qos trust ports	802.1Q-tagged ports and mobile ports are always trusted; any other port is untrusted
Statistics interval	qos stats interval	60 seconds
Global bridged disposition	qos default bridged disposition	accept
Global multicast disposition	qos default multicast disposition	accept
Global default DEI bit setting	qos dei	disabled
Level of log detail	qos log level	6
Number of lines in QoS log	qos log lines	256
Whether log messages are sent to the console	qos log console	no
Whether log messages are available to OmniVista applications	qos forward log	no
Whether IP anti-spoofing is enabled on UserPorts.	qos user-port filter	yes
Whether a UserPorts port is administratively disabled when unwanted traffic is received.	qos user-port shutdown	no
Automatic NMS traffic prioritization.	qos nms priority	enabled
Priority for IP Phone connections.	qos phones	priority 5
Type of messages logged	debug qos	info

QoS Port Defaults

Use the **qos port reset** command to reset port settings to the defaults.

Description	Command/keyword	Default
The default 802.1p value inserted into packets received on untrusted ports.	qos port default 802.1p	0
The default DSCP value inserted into packets received on untrusted ports.	qos port default dscp	0
The default egress priority value to use for packets received on trusted ports.	qos port default classification	DSCP
Whether the port uses strict priority or weighted fair queuing.	qos port servicing mode	strict priority queuing
The default maximum bandwidth for each of the eight CoS queues per port.	qos port q maxbw	maximum = port bandwidth
Whether the port is trusted or untrusted	qos port trusted	802.1Q and mobile ports are always trusted; other ports are untrusted
The maximum egress bandwidth	qos port maximum egress-bandwidth	port bandwidth
The maximum ingress bandwidth	qos port maximum ingress-bandwidth	port bandwidth
The Drop Eligible Indicator (DEI) bit setting.	qos port dei	disabled

Policy Rule Defaults

The following are defaults for the **policy rule** command:

Description	Keyword	Default
Policy rule enabled or disabled	enable disable	enabled
Determines the order in which rules are searched	precedence	0
Whether the rule is saved to flash immediately	save	enabled
Whether messages about flows that match the rule are logged.	log	no
How often to check for matching flow messages.	log interval	60 seconds
Whether to count bytes or packets that match the rule.	count	packets are counted

Description	Keyword	Default
Whether to send a trap for the rule.	trap	enabled (trap sent only on port disable action or UserPort shutdown operation).

Policy Action Defaults

The following are defaults for the **policy action** command:

Description	Keyword	Default
Whether the flow matching the rule should be accepted or denied	disposition	accept
Tri-Color Marking (TCM) mode		Single-rate TCM (srTCM) mode
- committed rate and burst size	cir cbs	CIR=0, CBS=10K
- peak rate and burst size	pir pbs	PIR=0, PBS=10K

Note that in the current software release, the **deny** and **drop** options produce the same effect that is, the traffic is silently dropped.

Note. There are no defaults for the **policy condition** command.

Default (Built-in) Policies

The switch includes some built-in policies, or default policies, for particular traffic types or situations where traffic does not match any policies. In all cases, the switch accepts the traffic and places it into default queues.

- *Other traffic*—Any traffic that does not match a policy is accepted or denied based on the global disposition setting on the switch. The global disposition is by default **accept**. Use the **qos default bridged disposition** and **qos default multicast disposition** commands to change the disposition as described in “Creating Policy Conditions” on page 34-33 and “Setting the Global Default Dispositions” on page 34-16.
- *The switch network group*—The switch has a default network group, called **switch**, that includes all IP addresses configured for the switch itself. This default network group may be used in policies. See “Creating Network Groups” on page 34-47 for more information about network groups.
- *Policy Port Groups*—The switch has built-in policy port groups for each slot. The groups are called **Slot01**, **Slot02**, etc. Use the **show policy port group** command to view the built-in groups.

QoS Configuration Overview

QoS configuration involves the following general steps:

1 Configuring Global Parameters. In addition to enabling/disabling QoS, global configuration includes settings such as global port parameters, default disposition for flows, and various timeouts. The type of parameters you might want to configure globally will depend on the types of policies you will be configuring. For example, if you want to set up policies for 802.1p or ToS/DSCP traffic, you may want to configure all ports as trusted ports.

Typically, you will not need to change any of the global defaults. See [“Global QoS Defaults” on page 34-12](#) for a list of the global defaults. See [“Configuring Global QoS Parameters” on page 34-16](#) for information about configuring global parameters.

2 Configuring QoS Port Parameters. This configuration includes setting up QoS parameters on a per port basis. Typically you will not need to change the port defaults. See [“QoS Port Defaults” on page 34-13](#) for a list of port defaults. See [“QoS Ports and Queues” on page 34-23](#) for information about configuring port parameters.

3 Setting Up Policies. Most QoS configuration involves setting up policies. See [“Creating Policies” on page 34-31](#).

4 Applying the Configuration. All policy rule configuration and some global parameters must be specifically applied through the `qos apply` command before they are active on the switch. See [“Applying the Configuration” on page 34-59](#).

Configuring Global QoS Parameters

This section describes the global QoS configuration, which includes enabling and disabling QoS, applying and activating the configuration, controlling the QoS log display, and configuring QoS port and queue parameters.

Enabling/Disabling QoS

By default QoS is enabled on the switch. If QoS policies are configured and applied, the switch will attempt to classify traffic and apply relevant policy actions.

To disable the QoS, use the **qos** command. For example:

```
-> qos disable
```

QoS is immediately disabled. When QoS is disabled globally, any flows coming into the switch are not classified (matched to policies).

To re-enable QoS, enter the **qos** command with the **enable** option:

```
-> qos enable
```

QoS is immediately re-enabled. Any policies that are active on the switch will be used to classify traffic coming into the switch.

Note that individual policy rules may be enabled or disabled with the **policy rule** command.

Setting the Global Default Dispositions

By default, bridged, routed, and multicast flows that do not match any policies are accepted on the switch. To change the global default disposition (which determines whether the switch will accept, deny, or drop the flow) for bridged and multicast flows, use the desired disposition setting (**accept**, **drop**, or **deny**) with the following commands: **qos default bridged disposition** or **qos default multicast disposition**.

In the current release, the **drop** and **deny** options produce the same result (flows are silently dropped; no ICMP message is sent).

For example, to deny any multicast flows that do not match policies, enter:

```
-> qos default multicast disposition deny
```

To activate the setting, enter the **qos apply** command. For more information about the **qos apply** command, see [“Applying the Configuration” on page 34-59](#).

The default disposition for routed flows is not configurable on a global basis for the switch. Policies may be set up to deny any routed traffic through the switch.

Typically, the disposition is only configured when you are using policies for Access Control Lists (ACLs).

Note that if you set **qos default bridged disposition** to **deny**, you effectively drop all Layer 2 traffic that does not match any policy. If you want to create ACLs to allow some Layer 2 traffic through the switch, you must configure two rules for each type of Layer 2 traffic, one for source and one for destination. For more information about ACLs, see [Chapter 35, “Configuring ACLs.”](#)

Setting the Global Default Servicing Mode

The servicing mode refers to the queuing scheme used to shape traffic on destination (egress) ports. There are three schemes available: one strict priority and two weighted fair queueing (WFQ) options. By default all switch ports are set to use strict priority queuing.

The **qos default servicing mode** command is used to set the default queuing scheme for all switch ports. For example, the following command selects **wrr**—a WFQ scheme that uses 8 weighted round robin (WRR) queues—as the default servicing mode:

```
-> qos default servicing mode wrr
```

For more information about the available queuing schemes and configuring the servicing mode for individual ports, see [“Prioritizing and Queue Mapping” on page 34-23](#).

Automatic QoS Prioritization

Automatic QoS prioritization refers to prioritizing certain subsets of switch traffic without having to configure a specific QoS policy to do the same for each type of traffic. This functionality is currently available for Network Management System (NMS) traffic and IP phone traffic.

This section describes how to configure the automatic prioritization of NMS and IP phone traffic. The status of automatic NMS and IP phone prioritization for the switch is displayed through the **show qos config** command. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuring Automatic Prioritization for NMS Traffic

Prioritizing NMS traffic destined for the switch helps to maximize NMS access to the switch and reduce the risk of DoS attacks. The following types of traffic are considered NMS traffic:

- SSH (TCP Port 22)
- Telnet (TCP Port 23)
- WebView (HTTP Port 80)
- SNMP (UDP port 161)

The **qos nms priority** command is used to enable or disable the automatic prioritization of NMS traffic. This functionality is enabled for the switch by default. To disable automatic prioritization, use the **no** form of the **qos nms priority** command. For example:

```
-> qos no nms priority
```

Note the following when configuring the status of automatic NMS traffic prioritization:

- Only the NMS traffic associated with the first eight *active* IP interfaces is prioritized; any such traffic from additional interfaces is not prioritized.
- The precedence of an active IP interface is determined by the value of the SNMP interface index (ifindex), which was assigned to the interface when it was created. The lower the ifindex value the higher the precedence; the higher the ifindex value the lower the precedence. Therefore, the eight IP interfaces with the lowest ifindex values are eligible for automatic prioritization of NMS traffic.
- To change the precedence of an IP interface, use the **ip interface ifindex** command and specify a higher (lower precedence) or lower (higher precedence) ifindex value.

- When automatic NMS prioritization is enabled, QoS policies that specify priority are not applied to the NMS traffic. Other QoS policies, however, are applied to this type of traffic as usual. If a policy specifies rate limiting, then the policy with the lowest rate limiting value is applied.

Configuring Automatic Prioritization for IP Phone Traffic

By default, the switch automatically sets the ingress priority value for IP phone traffic to 5. The egress priority of IP phone packets is set to the default priority value configured for the QoS port receiving such traffic.

IP phone traffic is detected by examining the source MAC address of the packet to determine if the address falls within the following ranges of IP phone MAC addresses:

```
00-80-9F-54-xx-xx to 00-80-9F-64-xx-xx
00-80-9F-66-xx-xx to 00-80-9F-6F-xx-xx.
```

In addition to prioritizing IP phone traffic, it is also possible to automatically prioritize non-IP phone traffic. This is done by adding up to four MAC addresses or four ranges of MAC addresses to the predefined QoS “alaPhone” MAC address group. See [“Creating MAC Groups” on page 34-50](#) for more information.

The **qos phones** command is used to enable or disable automatic prioritization of IP phone traffic. In addition, this command also applies a priority value to the traffic. For example, the following command specifies a priority value to apply for ingress IP phone traffic:

```
-> qos phones priority 1
```

To disable automatic IP phone traffic prioritization for the switch, enter the following command:

```
-> qos no phones
```

Note that when automatic prioritization of IP phone traffic is enabled, QoS policies that specify priority are not applied to the IP phone traffic. Other QoS policies, however, are applied to this type of traffic as usual. If a policy specifies rate limiting, then the policy with the lowest rate limiting value is applied.

Using the QoS Log

The QoS software in the switch creates its own log for QoS-specific events. You may modify the number of lines in the log or change the level of detail given in the log. The PolicyView application, which is used to create QoS policies stored on an LDAP server, may query the switch for log events; or log events can be immediately available to the PolicyView application via a CLI command. Log events may also be forwarded to the console in real time.

What Kind of Information Is Logged

The **debug qos** command controls what kind of information will be displayed in the log. The **qos log level** command determines how specific the log messages will be. See [“Log Detail Level” on page 34-19](#).

By default, only the most basic QoS information is logged. The types of information that may be logged includes rules, Layer 2 and Layer 3 information, etc. For a detailed explanation about the types of informa-

tion that may be logged, see the *OmniSwitch 6250 CLI Reference Guide*. A brief summary of the available keywords is given here:

debug qos keywords

info	mem	classifier
config	cam	sem
rule	mapper	pm
main	flows	ingress
route	queue	egress
hre	slot	nimsg
port	l2	
msg	l3	
sl		

To display information about any QoS rules on the switch, enter **debug qos rule**:

```
-> debug qos rules
```

To change the type of debugging, use **no** with the relevant type of information that you want to remove. For example:

```
-> debug qos no rule
```

To turn off debugging (which effectively turns off logging), enter the following:

```
-> no debug qos
```

Enter the **qos apply** command to activate the setting.

Number of Lines in the QoS Log

By default the QoS log displays a maximum of 256 lines. To change the maximum number of lines that may display, use the **qos log lines** command and enter the number of lines. For example:

```
-> qos log lines 30
```

The number of lines in the log is changed. To activate the change, enter the **qos apply** command.

Note. If you change the number of log lines, the QoS log may be completely cleared. To change the log lines without clearing the log, set the log lines in the **boot.cfg** file; the log will be set to the specified number of lines at the next reboot.

Log Detail Level

To change the level of detail in the QoS log, use the **qos log level** command. The log level determines the amount of detail that will be given in the QoS log. The **qos log level** command is associated with the **qos debug** command, which determines what kind of information will be included in the log.

The default log level is 6. The range of values is 1 (lowest level of detail) to 9 (highest level of detail). For example:

```
-> qos log level 7
```

The log level is changed immediately but the setting is not saved in flash. To activate the change, enter the **qos apply** command. For more information about the **qos apply** command, see [“Applying the Configuration” on page 34-59](#).

Note. A high log level value will impact the performance of the switch.

Forwarding Log Events

NMS applications may query the switch for logged QoS events. Use the **qos forward log** command to make QoS log events available to these applications in real time. For example:

```
-> qos forward log
```

To disable log forwarding, enter the following command:

```
-> qos no forward log
```

To activate the change, enter the **qos apply** command. For more information about the **qos apply** command, see [“Applying the Configuration” on page 34-59](#).

If event forwarding is disabled, NMS applications will still be able to query the QoS software for events, but the events will not be sent in real time.

Forwarding Log Events to the Console

QoS log messages may be sent to the switch logging utility, which is an event logging application available on the OmniSwitch. The configuration of the switch logging utility then determines if QoS messages are sent to a log file in the switch’s flash file system, displayed on the switch console, and/or sent to a remote syslog server.

To send log events to the switch logging utility, enter the following command:

```
-> qos log console
```

To disable immediate forwarding of events to switch logging, enter the following command:

```
-> qos no log console
```

To activate the change, enter the **qos apply** command. For more information about the **qos apply** command, see [“Applying the Configuration” on page 34-59](#).

Use the **swlog output** command to configure switch logging to output logging events to the console. Note that this is in addition to sending log events to a file in the flash file system of the switch. See the [“Using Switch Logging”](#) chapter in the *OmniSwitch 6250 Network Configuration Guide* for more information.

Displaying the QoS Log

To view the QoS log, use the **show qos log** command. The display is similar to the following:

```
**QOS Log**

Insert rule 0
Rule index at 0
Insert rule 1
Rule index at 1
Insert rule 2
Rule index at 2
Enable rule r1 (1) 1,1
Enable rule r2 (0) 1,1
Enable rule yuba1 (2) 1,1
Verify rule r1(1)
```

```
Enable rule r1 (1) 1,1
Really enable r1
Update condition c1 for rule 1 (1)
Verify rule r2(1)
Enable rule r2 (0) 1,1
Really enable r2
Update condition c2 for rule 0 (1)
Verify rule yubal(1)
Enable rule yubal (2) 1,1
Really enable yubal
Update condition yubamac for rule 2 (1)
QoS Manager started TUE MAR 10 13:46:50 2002

Match rule 2 to 1
Match rule 2 to 2
Match rule 2 to 3
```

The log display may be modified through the **qos log lines**, **qos log level**, and **debug qos** commands. The log display may also be output to the console through the **qos log console** command or sent to the policy software in the switch (which manages policies downloaded from an LDAP server) through the **qos forward log** command.

Clearing the QoS Log

The QoS log can get large if invalid rules are configured on the switch, or if a lot of QoS events have taken place. Clearing the log makes the file easier to manage.

To clear the QoS log, use the **qos clear log** command. For example:

```
-> qos clear log
```

All the current lines in the QoS log are deleted.

Classifying Bridged Traffic as Layer 3

In some network configurations you may want to force the switch to classify bridged traffic as routed (Layer 3) traffic. Typically this option is used for QoS filtering. See [Chapter 35, "Configuring ACLs,"](#) for more information about filtering.

The Layer 3 classification of bridged traffic is no different from the classification of normal Layer 3 routed traffic. Note that this implementation of QoS always performs Layer 3 classification of bridged traffic; it is not an option. As a result,

- Layer 3 ACLs are always effected on bridged traffic.
- The switch may bridge and route traffic to the same destination.
- Bridged IP packets are prioritized based on ToS, not 802.1p.

Note that Layer 3 ACLs are effected on bridged IP traffic and Layer 2 ACLs are effected on routed traffic.

Setting the Statistics Interval

To change how often the switch polls the network interfaces for QoS statistics, use the **qos stats interval** command with the desired interval time in seconds. The default is 60 seconds. For example:

```
-> qos stats interval 30
```

Statistics are displayed through the **show qos statistics** command. For more information about this command, see the *OmniSwitch 6250 CLI Reference Guide*.

Returning the Global Configuration to Defaults

To return the global QoS configuration to its default settings, use the **qos reset** command. The defaults will then be active on the switch. For a list of global defaults, see “QoS Defaults” on page 34-12.

Note. The **qos reset** command only affects the global configuration. It does not affect any policy configuration.

Verifying Global Settings

To display information about the global configuration, use the following **show** commands:

show qos config	Displays global information about the QoS configuration.
show qos statistics	Displays statistics about QoS events.

For more information about the syntax and displays of these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

QoS Ports and Queues

Queue parameters may be modified on a port basis. When a flow coming into the switch matches a policy, it is queued based on:

- Parameters given in the policy action (specified by the **policy action** command) with either of the following keywords: **priority**, **maximum bandwidth**, or **maximum depth**.
- Port settings configured through the **qos port** command.

Shared Queues

Eight priority queues are available at startup for each port. Flows always share queues; however, when a **shared** action is specified in policies, the policies will use the same values to implement maximum bandwidth.

Prioritizing and Queue Mapping

QoS prioritizes packets by placing them in a higher priority egress queue. As previously mentioned, there are eight egress queues available for each port. In addition, there are different queuing algorithms available for egressing packets of different priorities. The algorithm used is determined by the servicing mode that is active for the egress port. See [“Configuring the Servicing Mode for a Port” on page 34-26](#) for more information.

The egress priority of a packet is determined as follows:

- 1** If a packet matches a QoS policy rule that sets a priority value, the egress priority for the packet is set using the value specified in the rule.
- 2** If a packet ingressing on a *trusted* port does not match any QoS policy rule that sets the priority, then the egress priority for the packet is set using the existing DSCP value (IP packets), the existing 802.1p value (non-IP packets), or the default classification priority value for the port. See [“Configuring Trusted Ports” on page 34-29](#) for more information.
- 3** If the default classification priority value for the port is set to DSCP, the DSCP value of a tagged IP packet is mapped to the 802.1p value for that same packet. In other words, the 802.1p priority is overwritten with the DSCP value. This does not apply to Layer 2 packets. See [“Maintaining the 802.1p Priority for IP Packets” on page 34-24](#) for more information.
- 4** If a packet ingressing on a *trusted* port does not have an 802.1p value, the egress priority for the packet is set using the default 802.1p priority value configured for the port.
- 5** The egress priority for a packet ingressing on a VLAN Stacking port (a trusted port) is set using the existing 802.1p value or configured through an associated VLAN Stacking service.
- 6** If a packet ingressing on an *untrusted* port does not match any QoS rule that sets the priority, then the egress priority for the packet is set using the default 802.1p value configured for the port on which the packet was received. See [“Configuring the Egress Queue Maximum Bandwidth” on page 34-27](#) for more information.
- 7** Note that the 802.1p bit for tagged packets ingressing on *untrusted* ports is set with the default 802.1p value, which is configured using the **qos port default 802.1p** command. If the packet is untagged, however, then the DSCP bit is set with the default DSCP value, which is configured using the **qos port default dscp** command.

Use the following table to see how packets are directed to the appropriate queues:

Priority to Queue Mapping Table

802.1p	ToS/DSCP	Rule(action) Priority	OS6250 Queue
0	000xxx	0	0
1	001xxx	1	1
2	010xxx	2	2
3	011xxx	3	3
4	100xxx	4	4
5	101xxx	5	5
6	110xxx	6	6
7	111xxx	7	7

Maintaining the 802.1p Priority for IP Packets

When a tagged IP packet ingresses on a trusted port and the default classification priority for that port is set to DSCP (using the default DSCP value of 0), the DSCP value of the packet is mapped to the 802.1p value of the same packet. To avoid overwriting the 802.1p value in this scenario, configure an ACL as follows:

- 1** Create a port group to include all of the ports that QoS should trust.
- 2** Define policy conditions for the port group; one condition for each L2 priority (802.1p) value.
- 3** Define policy actions that will stamp the IP traffic with the L2 priority value.
- 4** Define policy rules using the conditions and actions created in Steps 2 and 3.
- 5** Do not globally trust all switch ports.

For example:

```

-> policy port group VoIP 1/4-6 1/8 2/3-5
-> policy condition p0 destination port group VoIP
-> policy condition p1 destination port group VoIP
-> policy condition p2 destination port group VoIP
-> policy condition p3 destination port group VoIP
-> policy condition p4 destination port group VoIP
-> policy condition p5 destination port group VoIP
-> policy condition p6 destination port group VoIP
-> policy condition p7 destination port group VoIP
-> policy action p0 802.1p 0
-> policy action p1 802.1p 1
-> policy action p2 802.1p 2
-> policy action p3 802.1p 3
-> policy action p4 802.1p 4
-> policy action p5 802.1p 5
-> policy action p6 802.1p 6
-> policy action p7 802.1p 7
-> policy rule p0 condition p0 action p0
-> policy rule p1 condition p1 action p1
-> policy rule p2 condition p2 action p2

```

```
-> policy rule p3 condition p3 action p3
-> policy rule p4 condition p4 action p4
-> policy rule p5 condition p5 action p5
-> policy rule p6 condition p6 action p6
-> policy rule p7 condition p7 action p7
-> qos apply
```

Note that for pure Layer 2 packets, trusted ports will retain the 802.1p value of the packet and queue the packets according to that priority value.

Configuring Queuing Schemes

There are four queuing schemes available for each switch port: one strict priority scheme and three weighted fair queuing (WFQ) schemes. By default the strict priority scheme is used and consists of eight priority queues (SPQ). All eight queues on the port are serviced strictly by priority. Lower priority traffic is dropped in the presence of higher priority traffic.

The following WFQ schemes are available:

- **WRR**—All queues participate in a weighted round robin scheme. Traffic is serviced from each queue based on the weight of the queue.
- **Priority-WRR**—A type of WRR scheme that combines Strict-Priority queues (zero weight) and WRR queues (non-zero weight).
- **DRR**—All queues participate in a deficit round robin scheme. Traffic is serviced from each queue based on the weight of the queue.

The weight of each of the WRR/DRR queues is a configurable value. Use the following guidelines to configure WRR/DRR queue weights:

- Weights are configured with a value between 0 and 15. The default weight for each WRR/DRR queue is set to one. Each queue can have a different weight value, and configuring these values in ascending or descending order is *not* required. When a queue is given a weight of 0, it is configured as a Strict-Priority queue.
- A Priority-WRR scheme is configured by assigning a weight of zero to one or more WRR queues to make them Strict-Priority queues and a non-zero weight to the other WRR queues.
- If there are multiple SPQs configured, the SPQs are scheduled according to their CoS queue number before any WFQs are scheduled.
- The weight assigned to a WRR queue designates the number of packets the queue sends out before the scheduler moves on to the next queue. For example, a queue weight of 10 sends out 10 packets at each interval.
- The weight assigned to a DRR queue determines the number of bytes that the queue will service. The higher the queue weight assigned to a DRR queue, the higher the percentage of traffic that is serviced by that queue. For example, a queue with a weight of three will send four times as much traffic as a queue with a weight of one.
- On OmniSwitch 6250 switches, each DRR weight value is associated with the following number of bytes: 1=2K, 2=4K, 3=6K, 4=8K, 5=10K, 6=12K, 7=14K, 8=16K, 9=18K, 10=20K, 11=22K, 12=24K, 13=26K, 14=28K, 15=30K. For example, if the configured DRR queue weights are 1 1 2 2 3 3 4 4, queues 1 and 2 will service up to 2K each, queues 3 and 4 will service up to 4K each, queues 5 and 6 will service up to 6K each, and queues 7 and 8 will service up to 8K.

The queuing scheme selected is the scheme that is used to shape traffic on destination (egress) ports and is referred to as the QoS servicing mode for the port. It is possible to configure a default servicing mode that will apply to all switch ports (see [“Setting the Global Default Servicing Mode” on page 34-17](#)) or configure the servicing mode on an individual port basis (see [“Configuring the Servicing Mode for a Port” on page 34-26](#)).

Note that the QoS servicing mode only applies to destination ports because it is at this point where traffic shaping is effected on the flows. In addition, different ports can use different servicing modes.

Configuring the Servicing Mode for a Port

The **qos port servicing mode** command is used to configure the queuing scheme for an individual port. For example, the following command selects the strict priority scheme for port 1/2:

```
-> qos port 1/2 servicing mode strict-priority
```

The following command selects the WRR scheme for port 1/8:

```
-> qos port 1/8 servicing mode wrr
```

In the above example, a weight for each of the eight WRR queues was not specified; therefore, the default value of 1 is used for each queue. The following example selects the WRR scheme for port 1/10 and assigns a weighted value to each queue:

```
-> qos port 1/10 servicing mode wrr 0 2 3 4 8 1 1 7
```

To reset the servicing mode for the port back to the global default mode, use the **default** parameter with this command and do not specify a queuing scheme. For example,

```
-> qos port 1/10 servicing mode default
```

The **qos default servicing mode** command is used to set the global default queuing scheme that is used for all ports. See [“Setting the Global Default Servicing Mode” on page 34-17](#) for more information.

Note the following when configuring the port servicing mode:

- Only five unique port servicing mode configurations are allowed per slot. Once these five are used up, the remainder of ports on the slot will use the default servicing mode.
- The WRR and DRR queuing schemes are mutually exclusive for the switch. Once any port is configured with one of these two schemes, all remaining ports must use the same scheme as well.
- The **qos port servicing mode** command overrides the default servicing mode configured with the **qos default servicing mode** command.
- Once the **qos port servicing mode** command is used on a port, this same command is required to make any additional mode changes for that port. If the port is changed back to the default servicing mode, however, this restriction is removed and the **qos default servicing mode** command is also allowed on the port.

Bandwidth Shaping

Bandwidth shaping is configured on a per port basis. Bandwidth policing is applied using QoS policies (see [“Port Groups and Maximum Bandwidth”](#) on page 34-52 and [“Policy Applications”](#) on page 34-62 for more information).

QoS supports configuring maximum bandwidth on ingress and egress ports. In addition, the maximum egress bandwidth is configurable on a per Class-of-Service (COS) queue basis for each port (see [“Configuring the Egress Queue Maximum Bandwidth”](#) on page 34-27 for more information).

To limit the ingress or egress bandwidth for a QoS port, use the **qos port maximum egress-bandwidth** or **qos port maximum ingress-bandwidth** commands. For example,

```
-> qos port 1/1 maximum egress-bandwidth 10M
-> qos port 1/1 maximum ingress-bandwidth 5M
```

Note the following when configuring the ingress or egress bandwidth limit for a port:

- Maximum bandwidth limiting is done using a granularity of 64K bps. Any value specified that is not a multiple of 64K is rounded up to the next highest multiple of 64K.
- The maximum bandwidth value cannot exceed the maximum bandwidth of the interface type associated with the port.
- Modifying the maximum bandwidth is most useful for low-bandwidth links.
- The bandwidth limit configured using the **qos port maximum egress-bandwidth** command takes precedence over an egress queue limit configured on the same port.

Configuring the Egress Queue Maximum Bandwidth

Configuring a maximum bandwidth value for each of the eight queues on an egress port is allowed. The bandwidth values are set to zero by default, which means that the port speed is used for the maximum bandwidth.

To configure the bandwidth values use the **qos port q maxbw** command. For example, the following command sets the maximum bandwidth for queue 8 on port 2/10 to 2k and 10k:

```
-> qos port 2/10 q8 maxbw 10k
```

Note that configuring the bandwidth values for different queues requires a separate command for each queue.

Setting the DEI Bit

The Drop Eligible Indicator (DEI) bit setting is applied to packets marked yellow (non-conforming) as the result of Tri-Color Marking (TCM) rate limiting. The TCM policier meters traffic based on user-configured packet rates and burst sizes and then marks the metered packets as green, yellow, or red based on the metering results. See [“Tri-Color Marking”](#) on page 34-64 for more information.

Yellow packets are assigned a high drop precedence, which means they are dropped first when the egress port queues become congested. If there is no congestion on the queues, however, yellow packets are retained and forwarded along to the next switch. When this occurs, the receiving switch does not know that the packet was marked yellow by the transmitting switch.

Setting the DEI bit for yellow egress packets ensures that the upstream switch is made aware that the packet was marked yellow. The upstream switch can then decide to drop the DEI marked packets first when the network is congested.

The switch may be set globally so that DEI bit marking and mapping is enabled for all ports. Individual ports may be configured to override the global setting

Configuring the DEI Bit Setting

By default, DEI bit marking (egress) is disabled on all switch ports. The DEI bit setting operation may be configured globally on the switch, or on a per-port basis.

To configure the global DEI bit setting operation to mark traffic egressing on QoS destination ports, use the **qos dei** command with the **egress** parameter option. For example:

```
-> qos dei egress
```

To configure the DEI bit operation for an individual port, use the **qos port dei** with the **egress** parameter option. For example:

```
-> qos port 1/10 dei egress
```

See the *OmniSwitch 6250 CLI Reference Guide* for more information about these commands.

Trusted and Untrusted Ports

By default switch ports are *not trusted*; that is, they do not recognize 802.1p or ToS/DSCP settings in packets of incoming traffic. When a port is not trusted, the switch sets the 802.1p or ToS/DSCP bits in incoming packets to the default 802.1p or DSCP values configured for that port.

The **qos port default 802.1p** and **qos port default dscp** commands are used to specify the default 802.1p and ToS/DSCP values. If no default is specified, then these values are set to zero.

Fixed ports that are configured for 802.1Q are always trusted, regardless of QoS settings. They cannot be configured as untrusted. For more information about configuring 802.1Q for fixed ports, see [Chapter 22, “Configuring 802.1Q.”](#)

Mobile ports are also always trusted; however, mobile ports may or may not accept Q-tagged traffic.

Note about mobile ports. Mobile ports are not Q-tagged in the same manner as fixed ports; however, a mobile port will join a VLAN if tagged traffic for that VLAN comes in on the mobile port and the **vlan mobile-tag** function is enabled for that VLAN. For more information about tagging mobile port traffic, see [Chapter 4, “Configuring VLANs.”](#)

Ports must be *both trusted and configured for 802.1Q* traffic in order to accept 802.1p traffic.

The following applies to ports that are trusted (for 802.1p traffic, the ports must also be able to accept 802.1Q packets):

- The 802.1p or ToS/DSCP value is preserved.
- If the incoming 802.1p or ToS/DSCP flow does not match a policy, the switch places the flow into a default queue and prioritizes the flow based on the 802.1p or ToS/DSCP value in the flow.

- If the incoming 802.1p or ToS/DSCP flow matches a policy, the switch queues the flow based on the policy action.
- If the incoming 802.1p flow does not contain an 802.1p value, the switch uses the default 802.1p value configured for the port to prioritize the flow.

The switch may be set globally so that all ports are trusted. Individual ports may be configured to override the global setting.

Configuring Trusted Ports

By default, all ports (except 802.1Q-tagged ports and mobile ports) are untrusted. The trust setting is configurable on a global basis for the switch or on a per-port basis.

To configure the global setting on the switch, use the **qos trust ports** command. For example:

```
-> qos trust ports
```

To configure individual ports as trusted, use the **qos port trusted** command with the desired slot/port number. For example:

```
-> qos port 3/2 trusted
```

The global setting is active immediately; however, the port setting requires **qos apply** to activate the change. See [“Applying the Configuration” on page 34-59](#) for information about the **qos apply** command.

Using Trusted Ports With Policies

Whether or not the port is trusted is important when classifying traffic with 802.1p bits. If the policy condition specifies 802.1p, the switch must be able to recognize 802.1p bits. (Note that the trusted port must also be 802.1Q-tagged as described in [“Setting the DEI Bit” on page 34-27](#).)

The 802.1p bits may be set or mapped to a single value using the **policy action 802.1p** command. In this example, the **qos port** command specifies that port 2 on slot 3 will be able to recognize 802.1p bits. A policy condition (**Traffic**) is then created to classify traffic containing 802.1p bits set to 4 ingress on port 2 on slot 3. The policy action (**SetBits**) specifies that the bits will be reset to 7 when the traffic egresses the switch. A policy rule called **Rule2** puts the condition and the action together.

```
-> qos port 3/2 trusted
-> policy condition Traffic source port 3/2 802.1p 4
-> policy action SetBits 802.1p 7
-> policy rule Rule2 condition Traffic action SetBits
```

To activate the configuration, enter the **qos apply** command. For more information about the **qos apply** command, see [“Applying the Configuration” on page 34-59](#).

For actions that set 802.1p bits, note that a limited set of policy conditions are supported. For information about which conditions may be used with an 802.1p action, see [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#).

Note. 802.1p mapping may also be set for Layer 3 traffic, which typically has the 802.1p bits set to zero.

Verifying the QoS Port and Queue Configuration

To display information about QoS ports and queues, use the following commands:

show qos port

Displays information about all QoS ports or a particular port.

show qos queue

Displays information for all QoS queues or only those queues associated with a particular slot/port.

See the *OmniSwitch 6250 CLI Reference Guide* for more information about the syntax and displays for these commands.

Creating Policies

This section describes how to create policies in general. For information about configuring specific types of policies, see [“Policy Applications” on page 34-62](#).

Basic commands for creating policies are as follows:

- [policy condition](#)
- [policy action](#)
- [policy rule](#)

This section describes generally how to use these commands. For additional details about command syntax, see the *OmniSwitch 6250 CLI Reference Guide*.

Note. A policy rule may include a policy condition or a policy action that was created through PolicyView rather than the CLI. But a policy rule, policy action, or policy condition may only be modified through the source that created it. For example, if an action was created in PolicyView, it may be included in a policy rule configured through the CLI, but it cannot be modified through the CLI.

Policies are not used to classify traffic until the **qos apply** command is entered. See [“Applying the Configuration” on page 34-59](#).

To view information about how the switch will classify particular condition parameters, use the **show policy classify** command. This is useful to test conditions before actually activating the policies on the switch. See [“Testing Conditions” on page 34-43](#).

Quick Steps for Creating Policies

Follow the steps below for a quick tutorial on creating policies. More information about how to configure each command is given in later sections of this chapter.

- 1** Create a policy condition with the **policy condition** command. For example:

```
-> policy condition cond3 source ip 10.10.2.3
```

Note. (Optional) Test the rule with the **show policy classify** command using information from the policy condition. For example:

```
-> show policy classify 13 source ip 10.10.2.3
```

This command displays information about whether or not the indicated parameter may be used to classify traffic based on policies that are configured on the switch.

- 2** Create a policy action with the **policy action** command. For example:

```
-> policy action action2 priority 7
```

- 3** Create a policy rule with the **policy rule** command. For example:

```
-> policy rule my_rule condition cond3 action action2
```

- 4** Use the **qos apply** command to apply the policy to the configuration. For example:

```
-> qos apply
```

Note. (Optional) To verify that the rule has been configured, use the **show policy rule** command. The display is similar to the following:

```

-> show policy rule
      Policy          From  Prec  Enab  Act  Refl  Log  Trap  Save
r1          cli      0  Yes  Yes  No   No  Yes  Yes
(L2/3):      cond1 -> action1
r2          cli      0  Yes  Yes  No   No  Yes  Yes
(L2/3):      cond2 -> action4
+r3         cli      0  Yes  Yes  No   No  Yes  Yes
(L2/3):      cond3 -> action2

```

This command displays information about whether or not the indicated parameter may be used to classify traffic based on policies that are configured on the switch. For more information about this display, see [“Verifying Policy Configuration” on page 34-42](#).

An example of how the example configuration commands might display when entered sequentially on the command line is given here:

```

-> policy condition cond3 source ip 10.10.2.3
-> policy action action2 priority 7
-> policy rule my_rule condition cond3 action action2
-> qos apply

```

ASCII-File-Only Syntax

When the **policy rule**, **policy condition**, and **policy action** commands as well as any of the condition group commands are configured and saved in an ASCII file (typically through the **snapshot** command), the commands included in the file will include syntax indicating the command’s origin. The origin specifies where the rule, condition, condition group, or action was created, either an LDAP server or the CLI (**from ldap** or **from cli**). For built-in QoS objects, the syntax displays as **from blt**. For example:

```

-> policy action A2 from ldap disposition accept

```

The **from** option is configurable (for LDAP or CLI only) on the command line; however, it is not recommended that a QoS object’s origin be modified. The **blt** keyword indicates built-in; this keyword cannot be used on the command line. For information about built-in policies and QoS groups, see [“How Policies Are Used” on page 34-4](#).

Creating Policy Conditions

This section describes how to create policy conditions in general. Creating policy conditions for particular types of network situations is described later in this chapter.

Note. Policy condition configuration is not active until the **qos apply** command is entered. See [“Applying the Configuration” on page 34-59](#).

To create or modify a policy condition, use the **policy condition** command with the keyword for the type of traffic you want to classify, for example, an IP address or group of IP addresses. In this example, a condition (**c3**) is created for classifying traffic from source IP address 10.10.2.1:

```
-> policy condition c3 source ip 10.10.2.1
```

There are many options for configuring a condition, depending on how you want the switch to classify traffic for this policy. An overview of the options is given here. Later sections of this chapter describe how to use the options in particular network situations.

Note. The group options in this command refer to groups of addresses, services, or ports that you configure separately through policy group commands. Rather than create a separate condition for each address, service, or port, use groups and attach the group to a single condition. See [“Using Condition Groups in Policies” on page 34-46](#) for more information about setting up groups.

More than one condition parameter may be specified. Some condition parameters are mutually exclusive. For supported combinations of condition parameters, see [“Condition Combinations” on page 34-7](#).

policy condition keywords (ingress and egress)

source ip	service	source mac
source ipv6	service group	destination mac
destination ip		source mac group
destination ipv6	ip protocol	destination mac group
source network group	icmptype	
destination network group	icmptype	source vlan
	ethertype	source vlan group
source ip port		destination vlan (multicast only)
destination ip port	ipv6	
source tcp port		source port
destination tcp port	802.1p	source port group
source udp port	tos	destination port (multicast only)
destination udp port	dscp	destination port group (multicast only)
tcpflags		
established		

Note. The **source ipv6**, **destination ipv6**, **ipv6**, **source port** and **source port group** condition keywords are not supported by egress policies.

The condition will not be active on the switch until you enter the **qos apply** command.

Removing Condition Parameters

To remove a classification parameter from the condition, use **no** with the relevant keyword. For example:

```
-> policy condition c3 no source ip
```

The specified parameter (in this case, a source IP address) will be removed from the condition (**c3**) at the next **qos apply**.

Note. You cannot remove all parameters from a policy condition. A condition must be configured with at least one parameter.

Deleting Policy Conditions

To remove a policy condition, use the **no** form of the command. For example:

```
-> no policy condition c3
```

The condition (**c3**) cannot be deleted if it is currently being used by a policy rule. If a rule is using the condition, the switch will display an error message. For example:

```
ERROR: c3 is being used by rule 'my_rule'
```

In this case, the condition will not be deleted. The condition (**c3**) must first be removed from the policy rule (**my_rule**). See [“Creating Policy Rules” on page 34-35](#) for more information about setting up rules.

If **c3** is not used by a policy rule, it will be deleted after the next **qos apply**.

Creating Policy Actions

This section describes how to configure policy actions in general. Creating policy actions for particular types of network situations is described later in this chapter.

To create or modify a policy action, use the **policy action** command with the desired action parameter. A policy action should specify the way traffic should be treated. For example, it might specify a priority for the flow, a source address to rewrite in the IP header, or it may specify that the flow may simply be dropped. For example:

```
-> policy action Block disposition drop
```

In this example, the action (**Block**) has a disposition of **drop** (disposition determines whether a flow is allowed or dropped on the switch). This action may be used in a policy rule to deny a particular type of traffic specified by a policy condition.

Note. Policy action configuration is not active until the **qos apply** command is entered. See [“Applying the Configuration” on page 34-59](#).

More than one action parameter may be specified. Some parameters may be mutually exclusive. In addition, some action parameters are only supported with particular condition parameters. For information about supported combinations of condition and action parameters, see [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#). See the *OmniSwitch 6250 CLI Reference Guide* for details about command syntax.

policy action keywords (ingress and egress)

disposition	802.1p
shared	dcsp
priority	map
permanent gateway ip	port-disable
maximum bandwidth	redirect port
maximum depth	redirect linkagg
cir cbs pir pbs	no-cache
tos	mirror

Note. The **permanent gateway ip**, **redirect port**, **redirect linkagg** and **mirror** action keywords are not supported by egress policies. If you combine **priority** with **802.1p**, **dscp**, **tos**, or **map**, in an action, the priority value is used to prioritize the flow.

Removing Action Parameters

To remove an action parameter or return the parameter to its default, use **no** with the relevant keyword.

```
-> policy action a6 no priority
```

This example removes the configured priority value from action **a6**. If any policy rule is using action **a6**, the default action will be to allow the flow classified by the policy condition.

The specified parameter (in this case, priority) will be removed from the action at the next **qos apply**.

Deleting a Policy Action

To remove a policy action, use the **no** form of the command.

```
-> no policy action a6
```

The action cannot be deleted if it is currently being used by a policy rule. If a rule is using the action, the switch will display an error message. For example:

```
ERROR: a6 is being used by rule 'my_rule'
```

In this case, the action will not be deleted. The action (**a6**) must first be removed from the policy rule (**my_rule**). See [“Creating Policy Rules” on page 34-35](#) for more information about setting up rules.

If **a6** is not used by a policy rule, it will be deleted after the next **qos apply**.

Creating Policy Rules

This section describes in general how to create or delete policy rules and rule parameters. See later sections of this chapter for more information about creating particular types of policy rules.

To create a policy rule, use the **policy rule** command and specify the name of the rule, the desired condition, and the desired action.

In this example, condition **c3** is created for traffic coming from IP address 10.10.8.9, and action **a7** is created to prioritize the flow. Policy rule **rule5** combines the condition and the action, so that traffic arriving on the switch from 10.10.8.9 will be placed into the highest priority queue.

```
-> policy condition c3 source ip 10.10.8.9
-> policy action a7 priority 7
-> policy rule rule5 condition c3 action a7
```

The rule (**rule5**) will only take effect after the **qos apply** command is entered. For more information about the **qos apply** command, see [“Applying the Configuration” on page 34-59](#).

The **policy rule** command may specify the following keywords:

policy rule keywords

precedence
validity period
save
log
log interval
count
trap

In addition, a policy rule may be administratively disabled or re-enabled using the **policy rule** command. By default rules are enabled. For a list of rule defaults, see [“Policy Rule Defaults” on page 34-13](#).

Information about using the **policy rule** command options is given in the next sections.

Configuring a Rule Validity Period

A validity period specifies the days and times during which a rule is in effect. By default there is no validity period associated with a rule, which means the rule is always active.

To configure the days, months, times, and/or time intervals during which a rule is active, use the **policy validity period** command. Once the validity period is defined, it is then associated with a rule using the **policy rule** command. For example, the following commands create a validity period named **vp01** and associate it with rule **r01**:

```
-> policy validity period vp01 hours 13:00 to 19:00 days monday friday
-> policy rule r01 validity period vp01
```

Note the following when using validity periods to restrict the times when a rule is active:

- Only one validity period is associated with a policy rule. Each time this command is entered with a validity period name specified, the existing period name is overwritten with the new one.
- A rule is only in effect when all the parameters of its validity period are true. In the above example, rule **r01** is only applied between 13:00 and 19:00 on Mondays and Fridays. During all other times and days, the rule is not applied.
- Software and hardware resources are allocated for rules associated with a validity period even if the validity period is not active. Pre-allocating the resources makes sure the rule can be enforced when the validity period becomes active.

Disabling Rules

By default, rules are enabled. Rules may be disabled or re-enabled through the **policy rule** command using the **disable** and **enable** options. For example:

```
-> policy rule rule5 disable
```

This command prevents **rule5** from being used to classify traffic.

Note that if **qos disable** is entered, the rule will not be used to classify traffic even if the rule is enabled. For more information about enabling/disabling QoS globally, see [“Enabling/Disabling QoS” on page 34-16](#).

Rule Precedence

The switch attempts to classify flows coming into the switch according to policy precedence. Only the rule with the highest precedence will be applied to the flow. This is true even if the flow matches more than one rule.

Precedence is particularly important for Access Control Lists (ACLs). For more details about precedence and examples for using precedence, see [Chapter 35, “Configuring ACLs.”](#)

How Precedence is Determined

When there is a conflict between rules, precedence is determined using one of the following methods:

- **Precedence value**—Each policy has a precedence value. The value may be user-configured through the **policy rule** command in the range from 0 (lowest) to 65535 (highest). (The range 30000 to 65535 is typically reserved for PolicyView.) By default, a policy rule has a precedence of 0.
- **Configured rule order**—If a flow matches more than one rule and both rules have the same precedence value, the rule that was *configured first* in the list will take precedence.

Specifying Precedence for a Particular Rule

To specify a precedence value for a particular rule, use the **policy rule** command with the precedence keyword. For example:

```
-> policy rule r1 precedence 200 condition c1 action a1
```

Saving Rules

The **save** option marks the policy rule so that the rule will be captured in an ASCII text file (using the **configuration snapshot** command) and saved to the working directory (using the **write memory** command or **copy running-config working** command). By default, rules are saved.

If the **save** option is removed from a rule, the **qos apply** command may activate the rule for the current session, but the rule will not be saved over a reboot. Typically, the **no save** option is used for temporary policies that you do not want saved in the switch configuration file.

To remove the **save** option from a policy rule, use **no** with the **save** keyword. For example:

```
-> policy rule rule5 no save
```

To reconfigure the rule as saved, use the **policy rule** command with the **save** option. For example:

```
-> policy rule rule5 save
```

For more information about the **configuration snapshot**, **write memory**, and **copy running-config working** commands, see the *OmniSwitch 6250 Switch Management Guide* and the *OmniSwitch 6250 CLI Reference Guide*.

For more information about applying rules, see [“Applying the Configuration” on page 34-59](#).

Logging Rules

Logging a rule may be useful for determining the source of firewall attacks.

To specify that the switch should log information about flows that match the specified policy rule, use the **policy rule** command with the **log** option. For example:

```
-> policy rule rule5 log
```

To stop the switch from logging information about flows that match a particular rule, use **no** with the **log** keyword. For example:

```
-> policy rule rule5 no log
```

When logging is active for a policy rule, a logging interval is applied to specify how often to look for flows that match the policy rule. By default, the interval time is set to 30 seconds. To change the log interval time, use the optional **interval** keyword with the **log** option. For example:

```
-> policy rule rule5 log interval 1500
```

Note that setting the log interval time to 0 specifies to log as often as possible.

Deleting Rules

To remove a policy rule, use the **no** form of the command.

```
-> no policy rule rule1
```

The rule will be deleted after the next **qos apply**.

Creating Policy Lists

A QoS policy list provides a method for grouping multiple policy rules together and applying the group of rules to specific types of traffic. The type of traffic to which a policy list is applied is determined by the type of list that is configured. There are two types of policy lists:

- **Default**—This list is always available on every switch and is not configurable. By default, a policy rule is associated with this list when the rule is created. All default list rules are applied to ingress traffic.
- **Egress**—When a list is configured as an egress policy list, all rules associated with that list are applied to traffic egressing on QoS destination ports.

To create an egress policy list, use the **policy list** command and specify the list type and the names of one or more existing QoS policy rules to add to the list. For example, the following commands create two policy rules and associates these rules with the **egress_rules** list:

```
-> policy condition c1 802.1p 5
-> policy action a1 disposition drop
-> policy rule r1 condition c1 action a1
-> policy condition c2 source ip 10.5.5.0
-> policy action a2 disposition accept
-> policy rule r2 condition c2 action a2
```

```
-> policy list egress_rules type egress rules r1 r2 enable
-> qos apply
```

By default, a policy list is enabled at the time the list is created. To disable or enable a policy list, use the following commands:

```
-> policy list egress_rules disable
-> policy list egress_rules enable
```

To remove an individual rule from an egress policy list, use the following command:

```
-> policy list egress_rules no r5
```

To remove an entire egress policy list from the switch configuration, use the following command:

```
-> no policy list egress_rules
```

Use the [show policy list](#) command to display the QoS policy rule configuration for the switch.

Guidelines for Configuring Policy Lists

Consider the following guidelines when configuring QoS policy rules and lists:

- Create policy rules first before attempting to create a list. The [policy list](#) command requires that the specified policy rules must already exist in the switch configuration. See [“Creating Policies” on page 34-31](#).
- Not all policy conditions and actions are supported within egress rules (rules that are members of an egress list). For more egress policy list guidelines, see [“Using Egress Policy Lists” on page 34-40](#).
- A rule may belong to the default list and an egress policy list at the same time. In addition, a rule may also belong to multiple lists of the same type. Each time a rule is assigned to a policy list, however, an instance of that rule is created. Each instance is allocated system resources.
- By default, QoS assigns rules to the default policy list. To exclude a rule from this list, use the **no default-list** option of the [policy rule](#) command when the rule is created. See [“Using the Default Policy List” on page 34-40](#) for more information.
- Up to 13 policy lists (including the default list) are supported per switch.
- If a rule is a member of multiple policy lists but one or more of these lists are disabled, the rule is still active for those lists that are enabled.
- If the QoS status of an individual rule is disabled, then the rule is disabled for all policy lists, even if a list to which the policy belongs is enabled.
- Policy lists are not active on the switch until the **qos apply** command is issued.

The following sections provide important information about using the default and egress policy lists. In addition, the [“Policy List Examples” on page 34-41](#) section provides additional configuration examples of policy rules and list types.

Using the Default Policy List

A default policy list always exists in the switch configuration. By default, a policy rule is added to this list at the time the rule is created. A rule remains a member of the default list even when it is subsequently assigned to additional lists.

Each time a rule is assigned to a list, an instance of that rule is created and allocated system resources. As a result, rules that belong to multiple lists create multiple instances of the same rule. One way to conserve resources is to remove a rule from the default policy list.

To exclude a rule from the default policy list, use the **no default-list** option of the **policy rule** command when the rule is created. For example:

```
-> policy rule r1 condition c1 action a1 no default-list
```

The **no default-list** option can also remove an existing rule from the default list. For example, the **r2** rule already exists in the switch configuration but was not excluded from the default list at the time the rule was created. The following command removes the rule from the default list:

```
-> policy rule r2 condition c1 action a1 no default-list
```

To add an existing rule to the default list, use the **default-list** parameter option of the policy rule command. For example:

```
-> policy rule r2 condition c1 action a1 default-list
```

Rules associated with the default policy list are applied only to ingress traffic, unless the rule is also assigned to an egress policy list.

Using Egress Policy Lists

Egress policy lists are used to direct QoS to apply policy rules to egress traffic. If a rule is not a member of an egress policy list, the rule only applies to ingress traffic.

An egress policy list is created using the **policy list** command and specifying **egress** as the policy type. For example:

```
-> policy list egress_rules type egress rules r1 r2 r3
```

The rules associated with an egress list are created in the same manner as all other policy rules. However, the following policy conditions and actions are not supported within egress rules:

- IPv6 conditions (any condition using the **ipv6** keyword).
- Source port and source port group conditions.
- Destination VLAN and destination VLAN group conditions.
- Internal priority/CoS actions.
- Tri-Color Marking (TCM) policy actions
- Port or linkagg redirect actions.
- Port disable, no caches, and permanent gateway IP actions.

See [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#) for more information about policy conditions and actions supported by both ingress and egress rules.

Consider the following additional guidelines for using egress policy lists:

- QoS changes DSCP and 802.1p values for traffic ingressing on an *untrusted* port. As a result, the new values may not match any egress policy list rules as expected. To avoid this scenario, trust the ingress port or configure a default ToS/DSCP/802.1p value as required.
- If an egress policy list rule contains an 802.1p condition and the ingress port is *trusted*, set the default classification of the ingress port to 802.1p. If the default classification of the ingress port is set to DSCP, the 802.1p value of the traffic is changed per the DSCP classification and will not match the egress 802.1p condition.
- An egress policy rule supports a maximum of two destination port groups.
- Egress policy lists and VLAN translation Service Access Point (SAP) configurations are mutually exclusive. The switch only allows whichever of these two features is configured first.
- Egress rate limiting configured through an Ethernet Service SAP profile takes precedence over egress rate limiting specified within a QoS egress policy list rule.
- If there are no system resources available to assign a rule to an ingress policy list (the default list), assigning that same rule to an egress list is not allowed.

Policy List Examples

The following examples illustrate how to create policy lists for ingress, egress, or both ingress and egress policy rules. The type of list determines the type of traffic to which the rule is applied. The default list applies rules to ingress traffic; the egress list applies rules to egress traffic.

Example 1: Default List - Ingress Rules

The following example creates a policy rule (**rule1**). This rule will apply only to ingress traffic because the rule is automatically assigned to the default policy list.

```
-> policy condition cond1 source mac 00:11:22:33:44:55 source vlan 100
-> policy action act1 disposition drop
-> policy rule rule1 condition cond1 action act1
-> qos apply
```

In this example, the **policy rule** command does *not* use the **no default-list** parameter, so the rule is automatically assigned to the default policy list. The default list always exists and is not configurable. As a result, the **policy list** command is not required to assign the rule to the default list.

Example 2: Egress List - Egress Rules

The following example creates two policy rules (**rule1** and **rule2**) and assigns these rules to an egress policy list. These rules will apply only to egress traffic.

```
-> policy condition cond1 source mac 00:11:22:33:44:55 source vlan 100
-> policy condition cond2 source ip 1.2.3.4
-> policy action act1 disposition drop
-> policy action act2 maximum bandwidth 1.00M
-> policy rule rule1 condition cond1 action act1 no default-list
-> policy rule rule2 condition cond2 action act2 no default-list
-> policy list egress_rules1 type egress rules rule1 rule2
-> qos apply
```

In this example, the **policy rule** commands use the **no default-list** parameter so that **rule1** and **rule2** are *not* assigned to the default policy list. The **policy list** command is then used to assign **rule1** and **rule2** to

the **egress_rules1** policy list. Because these two rules are assigned to the **egress_rules1** policy list and *not* the default list, the rules are applied only to egress traffic.

Example 3: Default List and Egress List - Ingress and Egress Rules

The following example creates and assigns policy rules to the default policy list and an egress policy list.

```
-> policy vlan group vlan_group3 3000 3100-3105
-> policy condition c1 source mac 00:11:22:33:44:55 source vlan 100
-> policy condition c2 source ip 1.2.3.4
-> policy condition c3 source port 1/1 destination port 2/23
-> policy condition c4 source vlan group vlan_group3
-> policy action a1 disposition drop
-> policy action a2 maximum bandwidth 1.00M
-> policy action a3 802.1p 5
-> policy rule rule1 condition c1 action a1
-> policy rule rule2 condition c2 action a2
-> policy rule rule3 condition c3 action a3
-> policy rule rule4 condition c4 action a2 no default-list
-> policy list egress_rules1 type egress rules r1 r4
-> qos apply
```

In this example, **rule1**, **rule2**, and **rule3** are assigned to the default policy list and **rule1** and **rule4** are assigned to the **egress_rules1** list. As a result, these rules are applied as follows:

- Rules **rule2** and **rule3** are applied only to ingress traffic because they are associated with the default policy list and *not* the **egress_rules1** policy list.
- Rule **rule1** is applied to both ingress and egress traffic because the rule is associated with both the default policy list *and* the **egress_rules1** policy list.
- Rule **rule4** is applied only to egress traffic because the rule is associated with the **egress_rules1** policy list and *not* the default list.

Verifying Policy Configuration

To view information about policy rules, conditions, and actions configured on the switch, use the following commands:

show policy rule	Displays information about all pending and applied policy conditions or a particular policy condition configured on the switch. Use the applied keyword to display information about applied conditions only.
show policy action	Displays information about all pending and applied policy actions or a particular policy action configured on the switch. Use the applied keyword to display information about applied actions only.
show policy rule	Displays information about all pending and applied policy rules or a particular policy rule. Use the applied keyword to display information about applied rules only.
show active policy rule	Displays applied policy rules that are active (enabled) on the switch.
show active policy rule meter-statistics	Displays the Tri-color Marking (TCM) counter color statistics for active policy rules. See “ Tri-Color Marking ” on page 34-64 for information.
show policy list	Displays information about pending and applied policy lists.

When the command is used to show output for all pending and applied policy configuration, the following characters may appear in the display:

character	definition
+	Indicates that the policy rule has been modified or has been created since the last qos apply .
-	Indicates the policy object is pending deletion.
#	Indicates that the policy object differs between the pending/applied objects.

For example:

```
-> show policy rule
                Policy      From Prec  Enab  Act  Refl  Log  Trap  Save
my_rule
{L2/3}:        cli  0Yes  Yes  No   No   Yes  Yes
+my_rule5
{L2/3}:        cli  0Yes  No   No   No   Yes  Yes
mac1
{L2/3}:        cli  0Yes  No   No   No   Yes  Yes
```

The above display indicates that **my_rule** is inactive and is not used to classify traffic on the switch (the Inact field displays **Yes**). The rule **my_rule5** has been configured since the last **qos apply** command was entered, as indicated by the plus (+) sign. The rule will not be used to classify traffic until the next **qos apply**. Only **mac1** is actively being used on the switch to classify traffic.

To display only policy rules that are active (enabled and applied) on the switch, use the **show active policy rule** command. For example:

```
-> show active policy rule
                Policy      From Prec  Enab  Act  Refl  Log  Trap  Save  Matches
mac1
{L2/3}:        cli  0   Yes  Yes  No   No  Yes  Yes   0
```

In this example, the rule **my_rule** does not display because it is inactive. Rules are inactive if they are administratively disabled through the **policy rule** command, or if the rule cannot be enforced by the current hardware. Although **my_rule5** is administratively active, it is still pending and not yet applied to the configuration. Only **mac1** is displayed here because it is active on the switch.

See the *OmniSwitch 6250 CLI Reference Guide* for more information about the output of these commands.

Testing Conditions

Before applying policies to the configuration through the **qos apply** command, you may want to see how the policies will be used to classify traffic. Or you may want to see how theoretical traffic would be classified by policies that are already applied on the switch.

Use the **policy action cir** commands to see how the switch will classify certain condition parameters. This command is used to examine the set of pending policies only. Use the **applied** keyword with the command to examine the applied set of policies only. The command includes a keyword (**l2**, **l3**, **multicast**) to indicate whether the Layer 2, Layer 3, or multicast classifier should be used to classify the traffic.

The keywords used with these commands are similar to the keywords used for the [policy condition](#) command. The keyword should be relevant to the type of traffic as listed in the table here:

show policy classify l2	show policy classify l3	
source port	source port	destination vlan (multicast only)
destination port	destination mac	destination ip
source mac	destination port	ip protocol
destination mac	source ip	tos
source vlan	source ipv6	dscp
	destination ip	802.1p
	destination ipv6	
	source ip port	
	destination ip port	

To test a theoretical condition against the set of pending policies, enter the command and the relevant keyword and value. The switch will display information about the potential traffic and attempt to match it to a policy (pending policies only). For example:

```
-> show policy classify l2 destination mac 08:00:20:d1:6e:51
Packet headers:
L2:
 *Port          :          0/0    ->    0/0
 *IfType        :          any    ->    any
 *MAC           :    000000:000000 ->    080020:D1E51
 *VLAN          :          0      ->    0
 *802.1p        :    0
L3/L4:
 *IP            :    0.0.0.0      ->    0.0.0.0
 *TOS/DSCP      :    0/0

Using pending l2 policies
Classify L2 Destination:
 *Matches rule 'yuba': action pri3 (accept)
Classify L2 Source:
 *No rule matched: (accept)
```

The display shows Layer 2 or Layer 3 information, depending on what kind of traffic you are attempting to classify. In this example, the display indicates that the switch found a rule, **yuba**, to classify destination traffic with the specified Layer 2 information.

To test a theoretical condition against the set of applied policies, enter the command with the **applied** keyword. The switch will display information about the potential traffic and attempt to match it to a policy (applied policies only). For example:

```
-> show policy classify l3 applied source ip 143.209.92.131 destination ip
198.60.82.5

Packet headers:
L2:
 *Port          :          0/0    ->    0/0
 *IfType        :          any    ->    any
 *MAC           :    000000:000000 ->    000000:000000
 *VLAN          :          0      ->    0
 *802.1p        :    0
L3/L4:
 *IP            :    143.209.92.131 ->    198.60.82.5
 *TOS/DSCP      :    0/0

Using applied l3 policies
Classify L3:
 *Matches rule 'r1': action a1 (drop)
```

In this example, the display indicates that the switch found an applied rule, **r1**, to classify Layer 3 flows with the specified source and destination addresses.

To activate any policy rules that have not been applied, use the **qos apply** command. To delete rules that have not been applied (and any other QoS configuration not already applied), use the **qos revert** command. See [“Applying the Configuration” on page 34-59](#).

Using Condition Groups in Policies

Condition groups are made up of multiple IPv4 addresses, MAC addresses, services, ports, or VLANs to which you want to apply the same action or policy rule. Instead of creating a separate condition for each address, etc., create a condition group and associate the group with a condition. Groups are especially useful when configuring filters, or Access Control Lists (ACLs); they reduce the number of conditions and rules that must be entered. For information about setting up ACLs, see [Chapter 35, “Configuring ACLs.”](#)

Commands used for configuring condition groups include the following:

```
policy network group
policy service group
policy mac group
policy port group
policy vlan group
```

ACLs

Access Control Lists (ACLs) typically use condition groups in policy conditions to reduce the number of rules required to filter particular types of traffic. For more information about ACLs, see [Chapter 35, “Configuring ACLs.”](#)

Sample Group Configuration

- 1 Create the group and group entries. In this example, a network group is created:

```
-> policy network group netgroup1 10.10.5.1 10.10.5.2
```

- 2 Attach the group to a policy condition. For more information about configuring conditions, see [“Creating Policy Conditions” on page 34-33.](#)

```
-> policy condition cond3 source network group netgroup1
```

Note. (Optional) Use the **show policy network group** command to display information about the network group. Each type of condition group has a corresponding show command. For example:

```
-> show policy network group
Group Name:          From      Entries
Switch              blt      4.0.1.166
                   10.0.1.166

+netgroup1          cli      10.10.5.1/255.255.255.0
                   10.10.5.2/255/255/255.0
```

See the *OmniSwitch 6250 CLI Reference Guide* for more information about the output of this display. See [“Verifying Condition Group Configuration” on page 34-55](#) for more information about using **show** commands to display information about condition groups.

3 Attach the condition to a policy rule. (For more information about configuring rules, see “[Creating Policy Rules](#)” on page 34-35.) In this example, action **act4** has already been configured. For example:

```
-> policy rule my_rule condition cond3 action act4
```

4 Apply the configuration. See “[Applying the Configuration](#)” on page 34-59 for more information about this command.

```
-> qos apply
```

The next sections describe how to create groups in more detail.

Creating Network Groups

Use network policy groups for policies based on IPv4 source or destination addresses. Note that IPv6 addresses are not supported with network groups at this time. The policy condition will specify whether the network group is a source network group, destination network group, or multicast network group.

- **Default switch group**—Note that by default the switch contains a network group called **switch** that includes all IPv4 addresses configured for the switch itself. This network group may also be used in policy conditions.
- **ACLs**—Typically network groups are used for Access Control Lists. For more information about ACLs, see [Chapter 35, “Configuring ACLs.”](#)

To create a network policy group, use the **policy network group** command. Specify the name of the group and the IPv4 address(es) to be included in the group. Each IPv4 address should be separated by a space. A mask may also be specified for an address. If a mask is not specified, the address is assumed to be a host address.

Note. Network group configuration is not active until the **qos apply** command is entered.

In this example, a policy network group called **netgroup2** is created with two IPv4 addresses. No mask is specified, so the IPv4 addresses are assumed to be host addresses.

```
-> policy network group netgroup2 10.10.5.1 10.10.5.2
```

In the next example, a policy network group called **netgroup3** is created with two IPv4 addresses. The first address also specifies a mask.

```
-> policy network group netgroup3 173.21.4.39 mask 255.255.255.0 10.10.5.3
```

In this example, the 173.201.4.39 address is subnetted, so that any address in the subnet will be included in the network group. For the second address, 10.10.5.3, a mask is not specified; the address is assumed to be a host address.

The network group may then be associated with a condition through the **policy condition** command. The network group must be specified as a **source network group** or **destination network group**. In this example, **netgroup3** is configured for condition **c4** as source network group:

```
-> policy condition c4 source network group netgroup3
```

To remove addresses from a network group, use **no** and the relevant address(es). For example:

```
-> policy network group netgroup3 no 173.21.4.39
```

This command deletes the 173.21.4.39 address from **netgroup3** after the next **qos apply**.

To remove a network group from the configuration, use the **no** form of the **policy network group** command with the relevant network group name. The network group must not be associated with any policy condition or action. For example:

```
-> no policy network group netgroup3
```

If the network group is not currently associated with any condition or action, the network group **netgroup3** is deleted from the configuration after the next **qos apply**.

If a condition or an action is using **netgroup3**, the switch will display an error message similar to the following:

```
ERROR: netgroup3 is being used by condition 'c4'
```

In this case, remove the network group from the condition first, then enter the **no** form of the **policy network group** command. For example:

```
-> policy condition c4 no source network group
-> no policy network group netgroup3
```

The **policy condition** command removes the network group from the condition. (See [“Creating Policy Conditions” on page 34-33](#) for more information about configuring policy conditions.) The network group will be deleted at the next **qos apply**.

Creating Services

Policy services are made up of TCP or UDP ports or port ranges. They include source or destination ports, or both, but the ports must be the same type (TCP *or* UDP). Mixed port types cannot be included in the same service.

Policy services may be associated with policy service groups, which are then associated with policy conditions; or they may be directly associated with policy conditions.

To create a service, use the **policy service** command. With this command, there are two different methods for configuring a service. You can specify the protocol and the IP port; or you can use shortcut keywords. The following table lists the keyword combinations:

Procedure	Keywords	Notes
Basic procedure for either TCP or UDP service	protocol source ip port destination ip port	<i>The protocol must be specified with at least one source or destination port.</i>
Shortcut for TCP service	source tcp port destination tcp port	<i>Keywords may be used in combination.</i>
Shortcut for UDP service	source udp port destination udp port	<i>Keywords may be used in combination.</i>

An IP protocol (TCP or UDP), source IP port and/or destination IP port (or port range) must be associated with a service. IP port numbers are well-known port numbers defined by the IANA. For example, port numbers for FTP are 20 and 21; Telnet is 23.

In this example, a policy service called **telnet1** is created with the TCP protocol number (**6**) and the well-known Telnet destination port number (**23**).

```
-> policy service telnet1 protocol 6 destination ip port 23
```

A shortcut for this command replaces the **protocol** and **destination ip port** keywords with **destination tcp port**:

```
-> policy service telnet1 destination tcp port 23
```

In the next example, a policy service called **ftp2** is created with port numbers for FTP (20 and 21):

```
-> policy service ftp2 protocol 6 source ip port 20-21 destination ip port 20
```

A shortcut for this command replaces the **protocol**, **source ip port**, and **destination ip port** keywords with **source tcp port** and **destination tcp port**:

```
-> policy service ftp2 source tcp port 20-21 destination tcp port 20
```

Multiple services created through the **policy service** command may be associated with a policy service group; or, individual services may be configured for a policy condition. If you have multiple services to associate with a condition, configure a service group and attach it to a condition. Service groups are described in [“Creating Service Groups” on page 34-49](#).

Note. Service configuration is not active until the **qos apply** command is entered.

To remove a policy service, enter the **no** form of the command.

```
-> no policy service ftp2
```

The **ftp2** service is deleted from the configuration at the next **qos apply** if the service is not currently associated with a policy condition or a service group.

Creating Service Groups

Service groups are made up of policy services. First configure the policy service, then create the service group which includes the policy service(s).

Use the **policy service group** command. For example:

```
-> policy service group serv_group telnet1 ftp2
```

In this example, a policy service group called **serv_group** is created with two policy services (**telnet1** and **ftp2**). The policy services were created with the **policy service** command. (See [“Creating Services” on page 34-48](#) for information about configuring policy services.)

Note. The policy service group can include only services with all source ports, all destination ports, or all source and destination ports. For example, the group cannot include a service that specifies a source port and another service that specifies a destination port.

The service group may then be associated with a condition through the **policy condition** command. For example:

```
-> policy condition c6 service group serv_group
```

This command configures a condition called **c6** with service group **serv_group**. All of the services specified in the service group will be included in the condition. (For more information about configuring conditions, see [“Creating Policy Conditions” on page 34-33.](#))

Note. Service group configuration must be specifically applied to the configuration with the **qos apply** command.

To delete a service from the service group, use **no** with the relevant service name. For example:

```
-> policy service group serv_group no telnet1
```

In this example, the service **telnet1** is removed from policy service group **serv_group**.

To delete a service group from the configuration, use the **no** form of the **policy service group** command. The service group must not be associated with any condition. For example:

```
-> no policy service group serv_group
```

Service group **serv_group** will be deleted at the next **qos apply**. If **serv_group** is associated with a policy condition, an error message will display instead. For example:

```
ERROR: serv_group is being used by condition 'c6'
```

In this case, remove the service group from the condition first; then enter the **no policy service group** command. For example:

```
-> policy condition c6 no service group
-> no policy service group serv_group
```

The **policy condition** command removes the service group from the policy condition. (See [“Creating Policy Conditions” on page 34-33](#) for more information about configuring policy conditions.) The service group will be deleted at the next **qos apply**.

Creating MAC Groups

MAC groups are made up of multiple MAC addresses that you want to attach to a condition.

To create a MAC group, use the **policy mac group** command.

For example:

```
-> policy mac group macgrp2 08:00:20:00:00:00 mask ff:ff:ff:00:00:00
00:20:DA:05:f6:23
```

This command creates MAC group **macgrp2** with two MAC addresses. The first address includes a MAC address mask, so that any MAC address starting with 08:00:20 will be included in **macgrp2**.

The MAC group may be then be associated with a condition through the **policy condition** command. Note that the policy condition specifies whether the group should be used for *source* or *destination*. For example:

```
-> policy condition cond3 source mac group macgrp2
```

This command creates a condition called **cond3** that may be used in a policy rule to classify traffic by source MAC addresses. The MAC addresses are specified in the MAC group. For more information about configuring conditions, see [“Creating Policy Conditions” on page 34-33.](#)

Note. MAC group configuration is not active until the **qos apply** command is entered.

To delete addresses from a MAC group, use **no** and the relevant address(es):

```
-> policy mac group macgrp2 no 08:00:20:00:00:00
```

This command specifies that MAC address 08:00:20:00:00:00 will be deleted from **macgrp2** at the next **qos apply**.

To delete a MAC group, use the **no** form of the **policy mac group** command with the relevant MAC group name. The group must not be associated with any policy condition. For example:

```
-> no policy mac group macgrp2
```

MAC group **macgrp2** will be deleted at the next **qos apply**. If **macgrp2** is associated with a policy condition, an error message will display instead:

```
ERROR: macgrp2 is being used by condition 'cond3'
```

In this case, remove the MAC group from the condition first; then enter the **no policy mac group** command. For example:

```
-> policy condition cond3 no source mac group
-> no policy mac group macgrp2
```

The **policy condition** command removes the MAC group from the condition. See [“Creating Policy Conditions” on page 34-33](#) for more information about configuring policy conditions. The MAC group will be deleted at the next **qos apply**.

Creating Port Groups

Port groups are made up of slot and port number combinations. Note that there are many built-in port groups, one for each slot on the switch. Built-in port groups are subdivided by slice. The built in groups are named by slot (**Slot01**, **Slot02**, etc.). To view the built-in groups, use the **show policy port group** command.

To create a port group, use the **policy port group** command. For example:

```
-> policy port group techpubs 2/1 3/1 3/2 3/3
```

The port group may then be associated with a condition through the **policy condition** command. Note that the policy condition specifies whether the group should be used for *source* or *destination*. For example:

```
-> policy condition cond4 source port group techpubs
```

This command creates a condition called **cond4** that may be used in a policy rule to classify traffic by source port number. The port numbers are specified in the port group. For more information about configuring conditions, see [“Creating Policy Conditions” on page 34-33](#).

Note. Port group configuration is not active until the **qos apply** command is entered.

To delete ports from a port group, use **no** and the relevant port number(s).

```
-> policy port group techpubs no 2/1
```

This command specifies that port 2/1 will be deleted from the **techpubs** port group at the next **qos apply**.

To delete a port group, use the **no** form of the **policy port group** command with the relevant port group name. The port group must not be associated with any policy condition. For example:

```
-> no policy port group techpubs
```

The port group **techpubs** will be deleted at the next **qos apply**. If **techpubs** is associated with a policy condition, an error message will display instead:

```
ERROR: techpubs is being used by condition 'cond4'
```

In this case, remove the port group from the condition first; then enter the **no policy port group** command. For example:

```
-> policy condition cond4 no source port group  
-> no policy port group techpubs
```

The **policy condition** command removes the port group from the policy condition. (See [“Creating Policy Conditions” on page 34-33](#) for more information about configuring policy conditions.) The port group will be deleted at the next **qos apply**.

Port Groups and Maximum Bandwidth

Maximum bandwidth policies are applied to source (ingress) ports and/or flows. If a port group condition is used in the policy, the bandwidth value specified is shared across all ports in the group. This also applies to flows that involve more than one port. For example, if a policy specifies a maximum bandwidth value of 10M for a port group containing 4 ports, the total bandwidth limit enforced is 10M for all 4 ports.

Note the following when configuring ingress maximum bandwidth policies:

- If a policy condition applies to ports that are located on different slots, the maximum bandwidth limit specified is multiplied by the number of slots involved. For example, if a rule is configured to apply a maximum bandwidth limit of 10M to ports 1/1, 3/10, and 4/5, then the actual bandwidth limit enforced for all three ports is 30M.
- The maximum traffic received by a destination port is also dependant on how many slots are sending traffic to the destination port. However, each slot is restricted to sending only 10k.
- If a policy condition applies to ports that are all on the same slot, then the maximum bandwidth value specified in the rule is not increased.
- Ingress bandwidth limiting is done using a granularity of 64K bps.
- The **show active policy list** command displays the number of packets that were dropped because they exceeded the ingress bandwidth limit applied by a maximum bandwidth policy.

The following example configures an ingress maximum bandwidth policy using a source port group.

Example: Source Port Group

In the following example, a port group (**pgroup**) is created with two ports and attached to a policy condition (**Ports**). A policy action with maximum bandwidth is created (**MaxBw**). The policy condition and policy action are combined in a policy rule called **PortRule**.

```
-> policy port group pgroup 1/1-2
-> policy condition Ports source port group pgroup
-> policy action MaxBw maximum bandwidth 10k
-> policy rule PortRule condition Ports action MaxBw
```

In this example, if both ports 1 and 2 are active ports, the 10000 bps maximum bandwidth is shared by both ports. In other words, maximum bandwidth policies for port groups define a maximum bandwidth value that is a total bandwidth amount for all ports, not an amount for each port.

Creating VLAN Groups

VLAN groups are made up of multiple VLAN IDs that you want to attach to a condition.

To create a VLAN group, use the **policy vlan group** command.

For example:

```
-> policy vlan group vlangrp1 10 15 20-25
```

This command creates VLAN group **vlangrp1** with two VLAN IDs and a range of VLAN IDs. This group may then be associated with a condition through the **policy condition** command. For example:

```
-> policy condition cond3 source vlan group vlangrp1
```

This command creates a condition called **cond3** that may be used in a policy rule to classify traffic by source VLAN IDs. The VLAN IDs are specified in the VLAN group. For more information about configuring conditions, see [“Creating Policy Conditions” on page 34-33](#).

Note. VLAN group configuration is not active until the **qos apply** command is entered.

To delete VLAN IDs from a VLAN group, use **no** and the relevant address(es):

```
-> policy mac group vlangrp1 no 15
```

This command specifies that VLAN ID 15 will be deleted from **vlangrp1** at the next **qos apply**.

When deleting a VLAN ID that falls within a specified range of VLAN IDs for the group, the entire range must be deleted. For example, to delete VLAN 23 from the group, the range 20-25 is specified:

```
-> policy mac group vlangrp1 no 20-25
```

This command specifies that VLAN IDs 20, 21, 22, 23, 24, and 25 will be deleted from **vlangrp1** at the next **qos apply**.

To delete a VLAN group, use the **no** form of the **policy vlan group** command with the relevant VLAN group name. The group must not be associated with any policy condition. For example:

```
-> no policy vlan group vlangrp1
```

VLAN group **vlangrp1** will be deleted at the next **qos apply**. If **vlangrp1** is associated with a policy condition, an error message will display instead:

```
ERROR: vlangrp1 is being used by condition 'cond3'
```

In this case, remove the VLAN group from the condition first; then enter the **no policy vlan group** command. For example:

```
-> policy condition cond3 no source vlan group
-> no policy vlan group vlangrp1
```

The **policy condition** command removes the VLAN group from the condition. See [“Creating Policy Conditions” on page 34-33](#) for more information about configuring policy conditions. The MAC group will be deleted at the next **qos apply**.

Verifying Condition Group Configuration

To display information about condition groups, use the following **show** commands:

show policy network group	Displays information about all pending and applied policy network groups or a particular network group. Use the applied keyword to display information about applied groups only.
show policy service	Displays information about all pending and applied policy services or a particular policy service configured on the switch. Use the applied keyword to display information about applied services only.
show policy service group	Displays information about all pending and applied policy service groups or a particular service group. Use the applied keyword to display information about applied groups only.
show policy mac group	Displays information about all pending and applied MAC groups or a particular policy MAC group configured on the switch. Use the applied keyword to display information about applied groups only.
show policy port group	Displays information about all pending and applied policy port groups or a particular port group. Use the applied keyword to display information about applied groups only.
show policy vlan group	Displays information about all pending and applied policy VLAN groups or a particular VLAN group. Use the applied keyword to display information about applied groups only.

See the *OmniSwitch 6250 CLI Reference Guide* for more information about the syntax and output for these commands.

When the command is used to show output for all pending and applied condition groups, the following characters may appear in the display:

character	definition
+	Indicates that the policy rule has been modified or has been created since the last qos apply .
-	Indicates the policy object is pending deletion.
#	Indicates that the policy object differs between the pending/applied objects.

In the example shown here, **netgroup1** is a new network group that has not yet been applied to the configuration.

```
-> show policy network group
Group Name:      From      Entries
Switch          blt      4.0.1.166
                10.0.1.166
                143.209.92.166
                192.85.3.1

+netgroup1      cli      143.209.92.0/255.255.255.0
                172.28.5.0/255/255/255.0
```

When the **qos apply** command is entered, the plus sign (+) will be removed from **netgroup1** in the display. See [“Applying the Configuration” on page 34-59](#) for more information about the **qos apply** command.

Using Map Groups

Map groups are used to map 802.1p, ToS, or DSCP values to different values. The following mapping scenarios are supported:

- 802.1p to 802.1p, based on Layer 2, Layer 3, and Layer 4 parameters and source/destination slot/port. In addition, 802.1p classification can trigger this action.
- ToS or DSCP to 802.1p, based on Layer 3 and Layer 4 parameters and source/destination slot/port. In addition ToS or DSCP classification can trigger this action.

Note. Map groups are associated with a policy *action*.

Commands used for creating map groups include the following:

policy map group
policy action map

Sample Map Group Configuration

1 Create the map group with mapping values. For detailed information about map groups and how to set them up, see [“How Map Groups Work”](#) on page 34-57 and [“Creating Map Groups”](#) on page 34-57.

```
-> policy map group tosGroup 1-2:5 4:5 5-6:7
```

2 Attach the map group to a policy action. See [“Creating Policy Actions”](#) on page 34-34 for more information about creating policy actions.

```
-> policy action tosMap map tos to 802.1p using tosGroup
```

Note. (Optional) Use the **show policy map group** command to verify the map group.

```
-> show policy map group
Group Name          From  Entries
+tosGroup           cli  1-2:5
                   4:5
                   5-6:7
```

For more information about this command, see [“Verifying Map Group Configuration”](#) on page 34-58 and the *OmniSwitch 6250 CLI Reference Guide*.

3 Attach the action to a policy rule. In this example, the condition **Traffic** is already configured. For more information about configuring rules, see [“Creating Policy Rules”](#) on page 34-35.

```
-> policy rule r3 condition Traffic action tosMap
```

4 Apply the configuration. For more information about this command, see [“Applying the Configuration”](#) on page 34-59.

```
-> qos apply
```

How Map Groups Work

When mapping from 802.1p to 802.1p, the action will result in remapping the specified values. Any values that are not specified in the map group are preserved. In this example, a map group is created for 802.1p bits.

```
-> policy map group Group2 1-2:5 4:5 5-6:7
-> policy action Map1 map 802.1p to 802.1p using Group2
```

The *to* and *from* values are separated by a colon (:). If traffic with 802.1p bits comes into the switch and matches a policy that specifies the **Map1** action, the bits will be remapped according to **Group2**. If the incoming 802.1p value is 1 or 2, the value will be mapped to 5. If the incoming 802.1p value is 3, the outgoing value will be 3 (the map group does not specify any mapping for a value of 3). If the incoming 802.1p value is 4, the value will be mapped to 5. If the incoming 802.1p value is 5 or 6, the value will be mapped to 7.

When mapping to a different type of value, however (ToS/DSCP to 802.1p), any values in the incoming flow that matches the rule but that are not included in the map group will be zeroed out. For example, the following action specifies the same map group but instead specifies mapping 802.1p to ToS:

```
-> policy action Map2 map tos to 802.1p using Group2
```

In this case, if ToS traffic comes into the switch and matches a policy that specifies the **Map2** action, the ToS value will be mapped according to **Group2** if the value is specified in **Group2**. If the incoming ToS value is 2, the value will be mapped to 5; however, if the incoming value is 3, the switch will map the value to zero because there is no mapping in **Group2** for a value of 3.

Note. Ports on which the flow is mapped must be a trusted port; otherwise the flow will be dropped.

Creating Map Groups

To create a map group, use the **policy action map** command. For example, to create a map group called **tosGroup**, enter:

```
-> policy map group tosGroup 1-2:5 4:5 5-6:7
```

The *to* and *from* values are separated by a colon (:). For example, a value of 2 will be mapped to 5.

Note. Map group configuration is not active until the **qos apply** command is entered.

The remapping group may then be associated with a rule through the **policy action** command. In this example, a policy condition called **Traffic** has already been configured.

```
-> policy action tosMap map tos to 802.1p using tosGroup
-> policy rule r3 condition Traffic action tosMap
```

To delete mapping values from a group, use **no** and the relevant values:

```
-> policy map group tosGroup no 1-2:4
```

The specified values will be deleted from the map group at the next **qos apply**.

To delete a map group, use the **no** form of the **policy map group** command. The map group must not be associated with a policy action. For example:

```
-> no policy map group tosGroup
```

If **tosGroup** is currently associated with an action, an error message similar to the following will display:

```
ERROR: tosGroup is being used by action 'tosMap'
```

In this case, remove the map group from the action, then enter the **no policy map group** command:

```
-> policy action tosMap no map group
-> no policy map group tosGroup
```

The map group will be deleted at the next **qos apply**.

Note. For Layer 2 flows, you cannot have more than one action that maps DSCP.

Verifying Map Group Configuration

To display information about all map groups, including all pending and applied map groups, use the **show policy map group** command. To display only information about applied map groups, use the **applied** keyword with the command. For more information about the output of this command, see the *OmniSwitch 6250 CLI Reference Guide*.

When the command is used to show output for all pending and applied condition groups, the following characters may appear in the display:

character	definition
+	Indicates that the policy rule has been modified or has been created since the last qos apply .
-	Indicates the policy object is pending deletion.
#	Indicates that the policy object differs between the pending/applied objects.

In the example here, a new map group, **tosGroup**, has not yet been applied to the configuration.

```
-> show policy map group
Group Name          From  Entries
+tosGroup           cli   1-2:5
                   4:5
                   5-6:7
```

When the **qos apply** command is entered, the plus sign (+) will be removed from **tosGroup** in the display. See [“Applying the Configuration” on page 34-59](#) for more information about the **qos apply** command.

Applying the Configuration

Configuration for policy rules and many global QoS parameters must specifically be applied to the configuration with the **qos apply** command. Any parameters configured without this command are maintained for the current session but are not yet activated. For example, if you configure a new policy rule through the **policy rule** command, the switch cannot use it to classify traffic and enforce the policy action until the **qos apply** command is entered. For example:

```
-> policy rule my_rule condition c4 action a5
-> qos apply
```

The **qos apply** command must be included in an ASCII text configuration file when QoS commands are included. The command should be included after the last QoS command.

When the configuration is not yet applied, it is referred to as the *pending configuration*.

Global Commands. Many global QoS commands are active immediately on the switch *without qos apply*. *The settings configured by these commands will be active immediately*. Other global commands must specifically be applied. The commands are listed in the following table:

Global Commands That Take Effect Immediately	Global Commands That Must Be Applied
qos qos forward log qos log console qos log lines qos log level debug qos qos trust ports qos stats interval qos revert qos flush qos reset	qos default bridged disposition qos default multicast disposition

Port and Policy Commands. All port parameters and policy parameters must be applied with the **qos apply** command.

Port and Policy Commands	
qos port policy condition policy action policy rule policy service policy service group	policy network group policy mac group policy port group policy vlan group policy map group

The pending configuration is useful for reviewing policy rules before actually applying them to the switch. The **show policy classify** commands may be used to review information about new conditions before they are applied on the switch. See [“Testing Conditions” on page 34-43](#).

Applied policy rules may also be administratively disabled (inactive). If a rule is administratively disabled, the rule will exist in the applied configuration but will not be used to classify flows. For more information about disabling/re-enabling a policy rule, see [“Creating Policy Rules” on page 34-35](#).

Deleting the Pending Configuration

Policy settings that have been configured but not applied through the **qos apply** command may be returned to the last applied settings through the **qos revert** command. For example:

```
-> qos revert
```

This command ignores any pending policies (any additions, modifications, or deletions to the policy configuration since the last **qos apply**) and writes the last applied policies to the pending configuration. At this point, the pending policies are the same as the last applied policies.

In this example, there are two new pending policies and three applied policies:

Pending Policies	Applied Policies
rule5	rule1
rule6	rule2
	rule3

If you enter **qos revert**, the configuration will then look like:

Pending Policies	Applied Policies
rule1	rule1
rule2	rule2
rule3	rule3

Flushing the Configuration

In some cases, you may want to remove all of your rules and start over again. To completely erase pending policies from the configuration, use the **qos flush** command. For example:

```
-> qos flush
```

If you then enter **qos apply**, all policy information will be deleted.

In this example, there are two new pending policies and three applied policies:

Pending Policies	Applied Policies
rule5	rule1
rule6	rule2
	rule3

If you enter **qos flush**, the configuration will then look like:

Pending Policies	Applied Policies
	rule1
	rule2
	rule3

In this scenario, you can do one of two things. To write the applied policies back to the pending configuration, use **qos revert**. Or, to delete all policy rule configuration, enter **qos apply**. If **qos apply** is entered, the empty set of pending policies will be written to the applied policies and all policy rule configuration will be deleted.

Interaction With LDAP Policies

The **qos apply**, **qos revert**, and **qos flush** commands do not affect policies created through the Policy-View application. Separate commands are used for loading and flushing LDAP policies on the switch. See [Chapter 31, “Managing Authentication Servers,”](#) for information about managing LDAP policies.

Verifying the Applied Policy Configuration

The policy **show** commands have an optional keyword (**applied**) to display only applied policy objects. These commands include:

show policy condition	Displays information about all pending and applied policy conditions or a particular policy condition configured on the switch. Use the applied keyword to display information about applied conditions only.
show policy action	Displays information about all pending and applied policy actions or a particular policy action configured on the switch. Use the applied keyword to display information about applied actions only.
show policy rule	Displays information about all pending and applied policy rules or a particular policy rule. Use the applied keyword to display information about applied rules only.
show policy network group	Displays information about all pending and applied policy network groups or a particular network group. Use the applied keyword to display information about applied groups only.
show policy service	Displays information about all pending and applied policy services or a particular policy service configured on the switch. Use the applied keyword to display information about applied services only.
show policy service group	Displays information about all pending and applied policy service groups or a particular service group. Use the applied keyword to display information about applied groups only.
show policy mac group	Displays information about all pending and applied MAC groups or a particular policy MAC group configured on the switch. Use the applied keyword to display information about applied groups only.
show policy port group	Displays information about all pending and applied policy port groups or a particular port group. Use the applied keyword to display information about applied groups only.
show policy vlan group	Displays information about pending and applied policy VLAN groups. Use the applied keyword to display information about applied groups only.
show policy map group	Displays information about all pending and applied policy map groups or a particular map group. Use the applied keyword to display information about applied groups only.
show policy classify	Sends Layer 2, Layer 3, or multicast information to the classifier to see how the switch will handle the packet. Use the applied keyword to examine only applied conditions.

For more information about these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Policy Applications

Policies are used to classify incoming flows and treat the relevant outgoing flows. There are many ways to classify the traffic and many ways to apply QoS parameters to the traffic.

Classifying traffic may be as simple as identifying a Layer 2 or Layer 3 address of an incoming flow. Treating the traffic might involve prioritizing the traffic or rewriting an IP address. How the traffic is treated (the *action* in the policy rule) typically defines the type of policy:

Type of Policy	Description	Action Parameters Used
Basic QoS policies	Prioritizes particular flows, and/or shapes the bandwidth for the flow	maximum bandwidth priority cir cbs pir pbs
Redirection policies	Redirects flows to a specific port or link aggregate ID.	redirect port redirect linkagg
Policy Based Mirroring	Mirrors ingress and egress packets to a specific port.	ingress mirror egress mirror ingress egress mirror
ICMP policies	Filters, prioritizes, and/or rate limits ICMP traffic	disposition priority maximum bandwidth
802.1p, ToS, and DSCP tagging or mapping policies	Sets or resets the egress 802.1p, ToS, or DSCP values	802.1p tos dscp map group
Policy Based Routing (PBR)	Redirects routed traffic.	permanent gateway ip
Access Control Lists (ACLs)	Groups of policies rules used for filtering traffic (allow/deny)	disposition

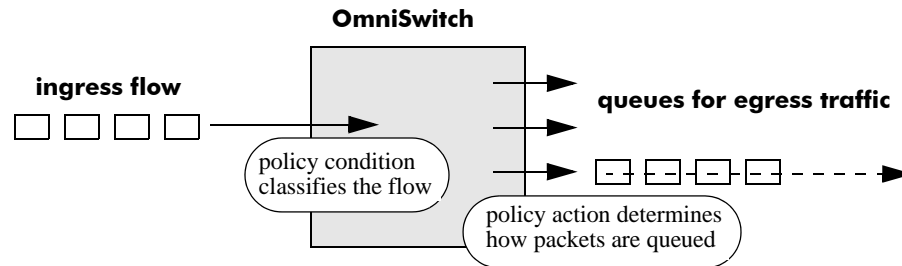
Note. The redirection policies, policy based mirroring, and policy based routing (PBR) are not supported by egress policies.

This section describes how to configure basic QoS policies and 802.1p/ToS/DSCP marking and mapping policies. Policies used for Layer 2 and Layer 3/4 filters, are commonly referred to as Access Control Lists (ACLs). Filtering is discussed in [Chapter 35, “Configuring ACLs.”](#)

Policies may also be used for prioritizing traffic in dynamic link aggregation groups. For more information about dynamic link aggregates, see [Chapter 24, “Configuring Dynamic Link Aggregation.”](#)

Basic QoS Policies

Traffic prioritization and bandwidth shaping may be the most common types of QoS policies. For these policies, any condition may be created; the policy action indicates how the traffic should be prioritized or how the bandwidth should be shaped.



Basic QoS Policy Application

Note. If multiple addresses, services, or ports should be given the same priority, use a policy condition group to specify the group and associate the group with the condition. See [“Using Condition Groups in Policies”](#) on page 34-46 for more information about groups.

Note that some condition parameters may be used in combination only under particular circumstances; also, there are restrictions on condition/action parameter combinations. See [“Using Condition Groups in Policies”](#) on page 34-46 and [“Condition Combinations”](#) on page 34-7.

Basic Commands

The following **policy action** commands are used for traffic prioritization or shaping:

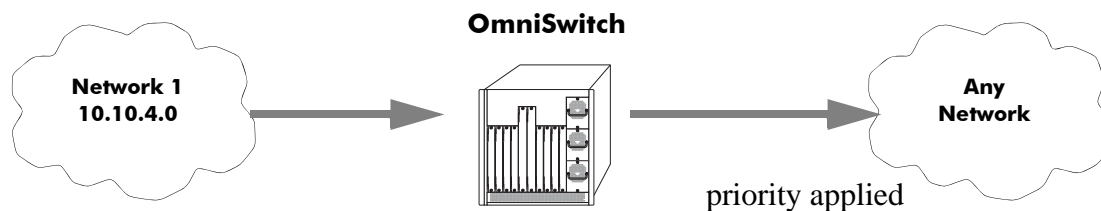
policy action priority
policy action maximum bandwidth

To set up traffic prioritization and/or bandwidth shaping, follow the steps in the next section. For more information about command syntax and options, see the *OmniSwitch 6250 CLI Reference Guide*.

Note that QoS ports may also be configured for bandwidth shaping through the **qos port** commands.

Traffic Prioritization Example

In this example, IP traffic is routed from the 10.10.4.0 network through the OmniSwitch.



Traffic Prioritization Example

To create a policy rule to prioritize the traffic from Network 1, first create a condition for the traffic that you want to prioritize. In this example, the condition is called **ip_traffic**. Then create an action to prioritize the traffic as highest priority. In this example, the action is called **high**. Combine the condition and the action into a policy rule called **rule1**.

```
-> policy condition ip_traffic source ip 10.10.4.0 mask 255.255.255.0
-> policy action high priority 7
-> policy rule rule1 condition ip_traffic action high
```

The rule is not active on the switch until the **qos apply** command is entered on the command line. When the rule is activated, any flows coming into the switch from 10.10.4.0 will be given the highest priority.

Bandwidth Shaping Example

In this example, a specific flow from a source IP address is sent to a queue that will support its maximum bandwidth requirement.

First, create a condition for the traffic. In this example, the condition is called **ip_traffic2**. A policy action (**flowShape**) is then created to enforce a maximum bandwidth requirement for the flow.

```
-> policy condition ip_traffic2 source ip 10.10.5.3
-> policy action flowShape maximum bandwidth 1k
-> policy rule rule2 condition traffic2 action flowShape
```

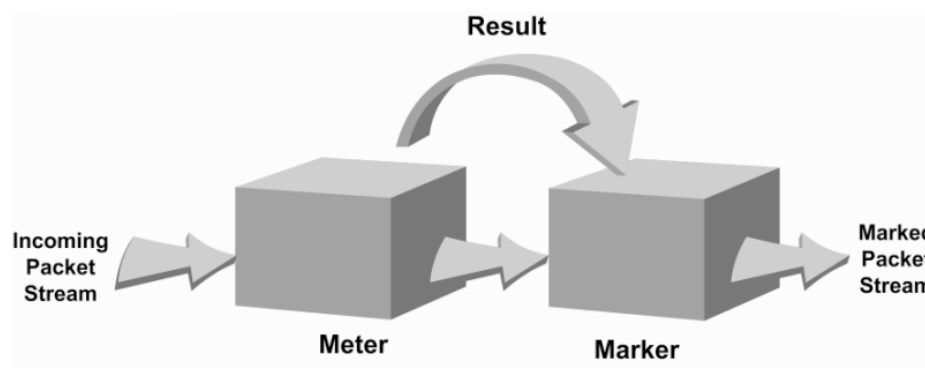
Note that the bandwidth may be specified in abbreviated units, in this case, **1k**.

The rule is not active on the switch until the **qos apply** command is entered. When the rule is activated, any flows coming into the switch from source IP address 10.10.5.3 will be queued with no more than 1k of bandwidth.

Tri-Color Marking

This implementation of a Tri-Color Marking (TCM) provides a mechanism for policing network traffic by limiting the rate at which traffic is sent or received on a switch interface. The TCM policier meters traffic based on user-configured packet rates and burst sizes and then marks the metered packets as green, yellow, or red based on the metering results.

The following diagram illustrates the basic operation of TCM:



The TCM policier meters each packet and passes the metering result along with the packet to the Marker. Depending upon the result sent by the Meter, the packet is then marked with either the green, yellow, or red color. The marked packet stream is then transmitted on the egress based on the color-coded priority assigned.

The TCM Meter operates in Color-Blind mode (the Color-Aware mode is not supported). In the Color-Blind mode, the Meter assumes that the incoming packet stream is uncolored.

There are two types of TCM marking supported:

- **Single-Rate TCM (srTCM)**—Packets are marked based on a Committed Information Rate (CIR) value and two associated burst size values: Committed Burst Size (CBS) and Peak Burst Size (PBS).
- **Two-Rate TCM (trTCM)**—Packets are marked based on a CIR value *and* a Peak Information Rate (PIR) value and two associated burst size values: CBS and PBS.

Both srTCM and trTCM operate in the same basic manner, as shown in the above diagram. The main difference between the two types is that srTCM uses one rate limiting value (CIR) and trTCM uses two rate limiting values (CIR and PIR) to determine packet marking.

The type of TCM used is determined when the policier is configured; depending on which rates and burst size values are configured, TCM will function in either single-rate or two-rate mode. There is no explicit command to select the type of TCM. See [“Configuring TCM Policies” on page 34-65](#) for more information.

Based on the TCM type used, packets are marked as follows:

TCM Type	Meter Compliance	Marker Color	Result
Single-Rate (srTCM)	Packet is CIR/CBS compliant.	GREEN	Packet is transmitted with the Drop Precedence set to LOW.
	Packet is not CIR/CBS compliant but is CIR/PBS compliant.	YELLOW	Packet is transmitted with the Drop Precedence set to HIGH (packet is dropped first when congestion occurs on the egress queue).
	Packet is neither CIR/CBS nor CIR/PBS compliant.	RED	Packet is dropped at the ingress.
Two-Rate (trTCM)	Packet is CIR/CBS compliant.	GREEN	Packet is transmitted with the Drop Precedence set to LOW.
	Packet is not CIR/CBS compliant but is PIR/PBS compliant.	YELLOW	Packet is transmitted with the Drop Precedence set to HIGH (packet is dropped first when congestion occurs on the egress queue).
	Packet is neither CIR/CBS nor PIR/PBS compliant.	RED	Packet is dropped at the ingress.

Configuring TCM Policies

Traffic rates and burst sizes used for TCM are configured using the following parameters in a QoS policy action or in a VLAN Stacking Service Access Point (SAP) profile:

- **cir** (Committed Information Rate, in bits per second)
- **cbs** (Committed Burst Size, in bytes)
- **pir** (Peak Information Rate, in bits per second)
- **pbs** (Peak Burst Size, in bytes)

For information about configuring these parameters for a VLAN Stacking SAP profile, see the “Configuring VLAN Stacking” chapter in this guide.

To configure a TCM QoS policy action, use the **policy action cir** command with one or more of the above parameters. Configuring the **cbs** and **pbs** parameters is optional. If a value is not specified for either one, the default value is used for both parameters. For example:

```
-> policy action A1 cir 10M
```

To specify one or both of the burst size values, use the **cbs** and **pbs** parameters. For example:

```
-> policy action A2 cir 10M cbs 4k
-> policy action A3 cir 10M cbs 4k pbs 10M
```

All of the above command examples configure the TCM meter to operate in the Single-Rate TCM (srTCM) mode. To configure the meter to operate in the Two-Rate TCM (trTCM) mode, use the **pir** parameter and specify a peak information rate value that is greater than the committed information rate value. For example, the following commands configure the meter to use the trTCM mode:

```
-> policy action A4 cir 10M cbs 4k pir 20M
-> policy action A5 cir 10M cbs 4k pir 20M pbs 40M
```

To remove the TCM configuration from a QoS policy action, use the **no** form of the **policy action cir** command. For example:

```
-> policy action A6 no cir
```

Consider the following when configuring TCM policy actions:

- There is no explicit CLI command to specify the mode in which the TCM meter operates. This mode is determined by whether or not the PIR is configured for the policy action and if the value of the PIR is greater than the value of the specified CIR. In this case, the trTCM mode is triggered; otherwise, the srTCM mode is used by default.
- This implementation of TCM is in addition to the basic rate limiting capabilities provided through the maximum bandwidth and maximum depth parameters used in QoS policy actions and the ingress bandwidth parameters used in VLAN Stacking Service Access Point (SAP) profiles. When these parameters are used, the TCM meter operates in the Single-Rate TCM mode by default.
- A srTCM policy action specifies both a CBS and PBS value. Default values for these burst sizes are used if one is not specified using the optional **cbs** and **pbs** parameters.
- Configure the PBS and CBS with a value that is greater than or equal to the size of the largest IP packet in the metered stream.

TCM Policy Example

Once configured, a TCM policy action is then available to use in a QoS policy rule to apply color marking to a specified traffic stream.

First, create a condition for the traffic. In this example, the condition is called **ip_traffic**. A policy action (**tcml**) is then created to enforce ingress rate limiting using TCM.

```
-> policy condition ip_traffic source ip 10.10.5.3
-> policy action tcml cir 10m cbs 4k pir 20m pbs 40m
-> policy rule rule1 condition ip_traffic action tcml
```

Note that the rates and burst sizes may be specified in abbreviated units, in this case, **10m**.

The rule is not active on the switch until the **qos apply** command is entered. When the rule is activated, any flows coming into the switch from source IP address 10.10.5.3 will be metered and marked according to the TCM policier parameters specified in the **tcml** policy action.

Redirection Policies

A redirection policy sends traffic that matches the policy to a specific port or link aggregate instead of the originally intended destination. This type of policy may use any condition; the policy action determines which port or link aggregate to which the traffic is sent.

The following **policy action** commands are used for port and link aggregate redirection:

```
policy action redirect port  
policy action redirect linkagg
```

Note that redirection policies apply to bridged traffic. When redirecting traffic on VLAN A, the redirect port or link aggregate ID must belong to VLAN A (tagged or default VLAN). In other words, the ingress port and redirect port must both reside in the same VLAN.

In the following example, flows destined for UDP port 80 is redirected to switch port 3/2:

```
-> policy condition L4PORTCOND destination udp port 80  
-> policy action REDIRECTPORT redirect port 3/2  
-> policy rule L4PORTRULE condition L4PORTCOND action REDIRECTPORT
```

In the following example, flows destined for IP address 40.2.70.200 are redirected to link aggregate 10:

```
-> policy condition L4LACOND destination IP 40.2.70.200  
-> policy action REDIRECTLA redirect linkagg 10  
-> policy rule L4LARULE condition L4LACOND action REDIRECTLA
```

Note that in both examples above, the rules are not active on the switch until the **qos apply** command is entered on the command line.

Policy-Based Mirroring

A mirroring policy sends a copy of ingress packets that match the policy condition to a specific port. This type of policy may use any condition; the mirror policy action determines the type of traffic to mirror and the port on which the mirrored traffic is received.

The **policy action mirror** command is used to configure mirror-to-port (MTP) action for the policy. For example, the following policy mirrors ingress packets to port 1/10:

```
-> policy condition c1 source ip 192.168.20.1  
-> policy action a1 mirror ingress 1/10  
-> policy rule r1 condition c1 action a1  
-> qos apply
```

When the above rule is activated, any flows coming into the switch from source IP address 192.168.20.1 are mirrored to port 1/10. It is also possible to combine the MTP action with other actions. For example:

```
-> policy condition c1 source ip 192.168.20.1  
-> policy action a1 mirror ingress 1/10 disposition drop  
-> policy rule r1 condition c1 action a1  
-> qos apply
```

This policy rule example combines the MTP action with the drop action. As a result, this rule drops ingress traffic with a source IP of 192.168.20.1, but the mirrored traffic from this source is not dropped and is forwarded to port 1/10.

Note the following regarding the use and configuration of mirroring policies:

- The policy source and destination ports must reside on different NI modules. Mirroring policies with the source and destination port on the same module are not supported.
- Only ingress policy-based mirroring is supported.
- Only one policy-based MTP session is supported at any given time. As a result, all mirroring policies should specify the same destination port.
- In addition to one policy-based MTP session, the switch can support one port-based mirroring session, one remote port mirroring (RPM) session, and one port monitoring session all running at the same time.
- If a packet qualifies for a policy-based MTP session and a port-based mirroring session (including remote port mirroring), the packet is copied to the destination port for both sessions.
- Policy based mirroring and the port-based mirroring feature can run simultaneously on the same port.
- Rule precedence is applied to all mirroring policies that are configured for the same switch ASIC. If traffic matches a mirror rule on one ASIC with a lower precedence than a non-mirroring rule on a different ASIC, the traffic is mirrored in addition to the actions specified by the higher precedence rule.
- Control PDUs are not mirrored.

ICMP Policy Example

Policies may be configured for ICMP on a global basis on the switch. ICMP policies may be used for security (for example, to drop traffic from the ICMP blaster virus).

In the following example, a condition called **icmpCondition** is created with no other condition parameters:

```
-> policy condition icmpCondition ip protocol 1
-> policy action icmpAction disposition deny
-> policy rule icmpRule condition icmpCondition action icmpAction
```

This policy (**icmpRule**) drops all ICMP traffic. To limit the dropped traffic to ICMP echo requests (pings) and/or replies, use the **policy condition icmptype** to specify the appropriate condition. For example,

```
-> policy condition echo icmptype 8
-> policy condition reply icmptype 0
```

802.1p and ToS/DSCP Marking and Mapping

802.1p values may be mapped to different 802.1p values on an individual basis or by using a map group. In addition, ToS or DSCP values may be mapped to 802.1p on a case-by-case basis or via a map group. (Note that any other mapping combination is not supported.)

Marking is accomplished with the following commands:

```
policy action 802.1p
policy action tos
policy action dscp
```

Mapping is accomplished through the following commands:

```
policy map group
policy action map
```

Note the following:

- Priority for the flow is based on the policy action. The value specified for 802.1p, ToS, DSCP, or the map group will determine how the flow is queued.
- The port on which the flow arrives (the ingress port) must be a trusted port. For more information about trusted ports, see [“Setting the DEI Bit” on page 34-27](#).

In this example, a policy rule (**marking**) is set up to mark flows from 10.10.3.0 with an 802.1p value of 5:

```
-> policy condition my_condition source ip 10.10.3.0 mask 255.255.255.0
-> policy action my_action 802.1p 5
-> policy rule marking condition my_condition action my_action
```

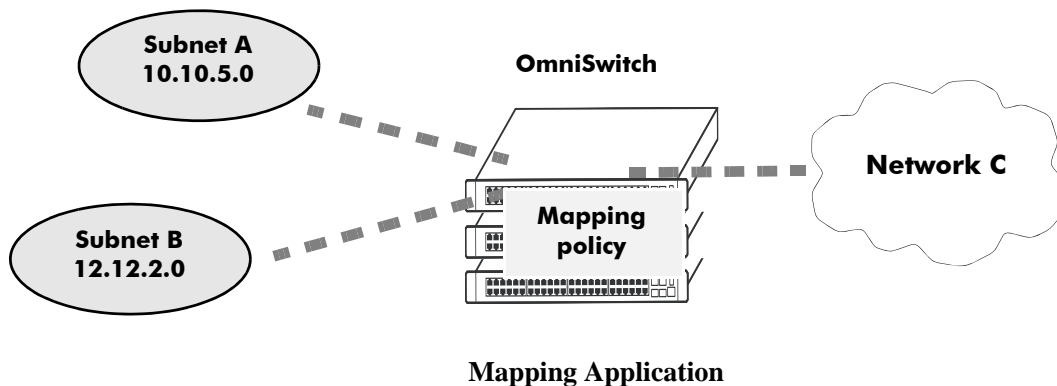
In the next example, the **policy map group** command specifies a group of values that should be mapped; the **policy action map** command specifies what should be mapped (802.1p to 802.1p, ToS/DSCP to 802.1p) and the mapping group that should be used. For more details about creating map groups, see [“Creating Map Groups” on page 34-57](#).

Here, traffic from two different subnets must be mapped to 802.1p values in a network called Network C. A map group (**tosGroup**) is created with mapping values.

```
-> policy map group tos_group 1-4:4 5-7:7
-> policy condition SubnetA source ip 10.10.5.0 mask 255.255.255.0
-> policy condition SubnetB source ip 12.12.2.0 mask 255.255.255.0
-> policy action map_action map tos to 802.1p using tos_group
```

The **map_action** specifies that ToS values will be mapped to 802.1p with the values specified in **tos_group**. With these conditions and action set up, two policy rules can be configured for mapping Subnet A and Subnet B to the ToS network:

```
-> policy rule RuleA condition SubnetA action map_action
-> policy rule RuleB condition SubnetB action map_action
```



Policy Based Routing

Policy Based Routing (PBR) allows a network administrator to define QoS policies that will override the normal routing mechanism for traffic matching the policy condition.

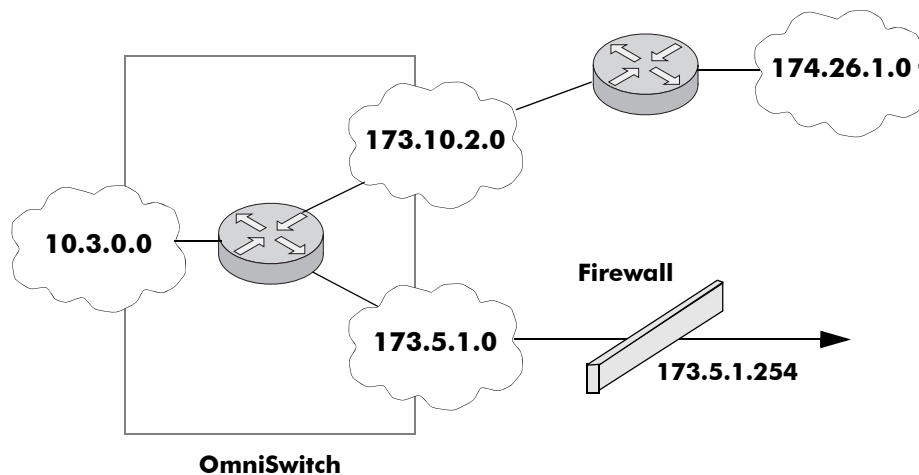
Note. When a PBR QoS rule is applied to the configuration, it is applied to the entire switch, unless you specify a built-in port group in the policy condition.

Policy Based Routing may be used to redirect traffic to a particular gateway based on source or destination IP address, source or destination network group, source or destination TCP/UDP port, a service or service group, IP protocol, or built-in source port group.

Traffic may be redirected to a particular gateway regardless of what routes are listed in the routing table. Note that the gateway address does not have to be on a directly connected VLAN; the address may be on any network that is learned by the switch.

Note. If the routing table has a default route of 0.0.0.0, traffic matching a PBR policy will be redirected to the route specified in the policy. For information about viewing the routing table, see [Chapter 25, “Configuring IP.”](#)

Policy Based Routing may be used to redirect untrusted traffic to a firewall. In this case, note that reply packets will not be allowed back through the firewall.



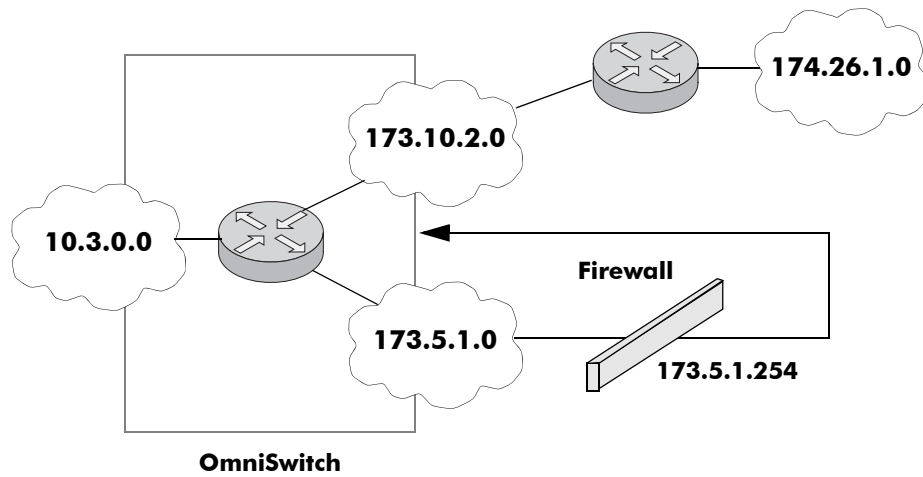
Routing all IP source traffic through a firewall

In this example, all traffic originating in the 10.3 network is routed through the firewall, regardless of whether or not a route exists.

```
-> policy condition Traffic3 source ip 10.3.0.0 mask 255.255.0.0
-> policy action Firewall permanent gateway ip 173.5.1.254
-> policy rule Redirect_All condition Traffic3 action Firewall
```

Note that the functionality of the firewall is important. In the example, the firewall is sending the traffic to be routed remotely. If you instead set up a firewall to send the traffic back to the switch to be routed, you should set up the policy condition with a built-in source port group so that traffic coming back from the firewall will not get looped and sent back out to the firewall.

For example:



Using a Built-In Port Group

In this scenario, traffic from the firewall is sent back to the switch to be re-routed. But because the traffic re-enters the switch through a port that is not in the Slot01 port group, the traffic does not match the Redirect_All policy and is routed normally through the switch.

```
-> policy condition Traffic3 source ip 10.3.0.0 mask 255.255.0.0 source port
group Slot01
-> policy action Firewall permanent gateway ip 173.5.1.254
-> policy rule Redirect_All condition Traffic3 action Firewall
```

Make sure to enter the **qos apply** command to activate the policy rule on the switch. Otherwise the rule will be saved as part of the pending configuration, but will not be active.

35 Configuring ACLs

Access Control Lists (ACLs) are Quality of Service (QoS) policies used to control whether or not packets are allowed or denied at the switch or router interface. ACLs are sometimes referred to as filtering lists.

ACLs are distinguished by the kind of traffic they filter. In a QoS policy rule, the type of traffic is specified in the policy condition. The policy action determines whether the traffic is allowed or denied. For detailed descriptions about configuring policy rules, see [Chapter 34, “Configuring QoS.”](#)

In general, the types of ACLs include:

- *Layer 2 ACLs*—for filtering traffic at the MAC layer. Usually uses MAC addresses or MAC groups for filtering.
- *Layer 3/4 ACLs*—for filtering traffic at the network layer. Typically uses IP addresses or IP ports for filtering.
- *Multicast ACLs*—for filtering IGMP traffic.

In This Chapter

This chapter describes ACLs and how to configure them through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- **Setting the Global Disposition.** The disposition specifies the general allow/deny policy on the switch. See [“Setting the Global Disposition” on page 35-7.](#)
- **Creating Condition Groups for ACLs.** Groups are used for filtering on multiple addresses, ports, or services. The group is then associated with the policy condition. See [“Creating Condition Groups For ACLs” on page 35-8.](#)
- **Creating Policy Rules for ACLs.** Policy rules for ACLs are basically QoS policy rules. Specific parameters for ACLs are described in this chapter. See [“Configuring ACLs” on page 35-9.](#)
- **Using ACL Security Features.** Specific port group, action, service group, and policy rule combinations are provided to help improve network security. See [“Using ACL Security Features” on page 35-15.](#)

ACL Specifications

The QoS/ACL functionality described in this chapter is supported on the OmniSwitch 6250 and OmniSwitch 6250-Metro Models switches unless otherwise stated in the following Specifications table or specifically noted within any other section of this chapter. Note that any maximum limits provided in the Specifications table are subject to available system resources.

Maximum number of policy rules	1280 (ingress and egress rules combined)
Maximum number of egress policy rules	512 (egress rules supported only on OmniSwitch 6250-Metro Models, Release 6.6.2.)
Maximum number of policy conditions	2048
Maximum number of policy actions	2048
Maximum number of policy validity periods	128
Maximum number of policy services	512
Maximum number of TCP and UDP port ranges	4
Maximum number of groups	1024 (VLAN groups supported only on OmniSwitch 6250-Metro Models, Release 6.6.2.)
Maximum number of group entries	1024 per group (512 per service group)
Maximum number of port groups per policy	8
Maximum number of rules per slot	1280
Maximum number of bandwidth shaping rules per slot	640
Maximum number of ToS or DSCP rules per slot	57
Maximum number of QoS policy lists per switch	13 (includes the default list)
Maximum number of priority queues per port	8
CLI Command Prefix Recognition	Some QoS commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch 6250 Switch Management Guide</i> for more information.

ACL Defaults

The following table shows the defaults for ACLs:

Parameter	Command	Default
Global bridged disposition	qos default bridged disposition	accept
Global multicast disposition	qos default multicast disposition	accept
Policy rule disposition	policy rule disposition	accept
Policy rule precedence	policy rule precedence	0 (lowest)

Note that in the current software release, the **deny** and **drop** options produce the same effect; that is, that traffic is silently dropped.

For more information about QoS defaults in general, see [Chapter 34, “Configuring QoS.”](#)

Quick Steps for Creating ACLs

1 Set the global disposition for bridged traffic. By default, all flows that do match any policies are allowed on the switch. However, you may want to deny traffic for all multicast flows that come into the switch and do not match a policy, but allow any Layer 2 (bridged) flows that do not match policies. For example:

```
-> qos default multicast disposition accept
```

2 Create policy condition groups for multiple addresses or services that you want to filter. (If you have a single address to filter, you can skip this step and simply include the address, service, or port in the policy condition.) An example:

```
-> policy network group NetGroup1 192.68.82.0 mask 255.255.255.0 192.60.83.0  
mask 255.255.255.0
```

3 Create a policy condition using the **policy condition** command. If you created a network group, MAC group, service group, or port group, specify the group as part of the condition.

```
-> policy condition Lab3 source network group NetGroup1
```

Note. (*Optional*) Test the condition with the **show policy classify** command using information from the policy condition. For example:

```
-> show policy classify l3 source ip 192.68.82.0
```

This command displays information about whether the indicated parameter may be used to classify traffic based on policies that are configured on the switch. For more information about testing conditions, see [“Testing Conditions” on page 34-43 in Chapter 34, “Configuring QoS.”](#)

4 Create a policy action with the **policy action** command. Use the keyword **disposition** and indicate whether the flow(s) should be accepted or denied.

```
-> policy action Yes disposition accept
```

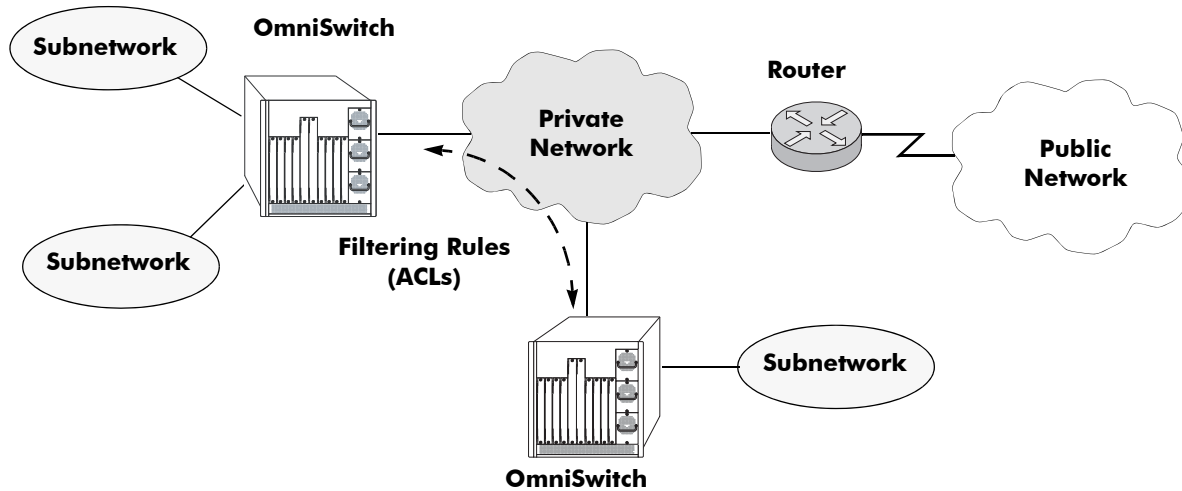
5 Create a policy rule with the **policy rule** command and include the relevant condition and action. Use the keyword **precedence** to specify the priority of this rule over other rules for traffic matching the specified condition.

```
-> policy rule lab_rule1 condition Lab3 action Yes precedence 65535
```

6 Apply the policy configuration using the **qos apply** command. For details about using this command, see [“Applying the Configuration” on page 34-59 in Chapter 34, “Configuring QoS.”](#)

ACL Overview

ACLs provide moderate security between networks. The following illustration shows how ACLs may be used to filter subnetwork traffic through a private network, functioning like an internal firewall for LANs.



Basic ACL Application

When traffic arrives on the switch, the switch checks its policy database to attempt to match Layer 2 or Layer 3/4 information in the protocol header to a filtering policy rule. If a match is found, it applies the relevant *disposition* to the flow. Disposition determines whether a flow is allowed or denied. There is a global disposition (the default is **accept**), and individual rules may be set up with their own dispositions.

Note. In some network situations, it is recommended that the global disposition be set to **deny**, and that rules be created to allow certain types of traffic through the switch. To set the global disposition to deny, use the **qos default bridged disposition** and **qos default multicast disposition** command. See [“Setting the Global Disposition”](#) on page 35-7 for more information about these commands.

When multiple policy rules exist for a particular flow, each policy is applied to the flow as long as there are no conflicts between the policies. If there is a conflict, then the policy with the highest precedence is applied to the flow. See [“Rule Precedence”](#) on page 35-6 for more information about precedence.

Note. QoS policy rules may also be used for traffic prioritization and other network scenarios. For a general discussion of QoS policy rules, see [Chapter 34, “Configuring QoS.”](#)

Rule Precedence

The switch attempts to classify flows coming into the switch according to policy precedence. Only the rule with the highest precedence will be applied to the flow. This is true even if the flow matches more than one rule.

How Precedence is Determined

When there is a conflict between rules, precedence is determined using one of the following methods:

- **Precedence value**—Each policy has a precedence value. The value may be user-configured through the **policy rule** command in the range from 0 (lowest) to 65535 (highest). (The range 30000 to 65535 is typically reserved for PolicyView.) By default, a policy rule has a precedence of 0.
- **Configured rule order**—If a flow matches more than one rule and both rules have the same precedence value, the rule that was *configured first* in the list will take precedence.

Interaction With Other Features

- **Routing Protocols**—Layer 3 filtering is compatible with routing protocols on the switch, including RIP.
- **Bridging**—Layer 2 and Layer 3 ACLs are supported for bridged and routed traffic. For information about classifying Layer 3 information in bridged frames, see [“Classifying Bridged Traffic as Layer 3” on page 34-21 in Chapter 34, “Configuring QoS.”](#)

Valid Combinations

There are limitations to the types of policy conditions and actions that may be combined in a single rule. For more information about supported combinations, see [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9 in Chapter 34, “Configuring QoS.”](#)

ACL Configuration Overview

This section describes the QoS CLI commands used specifically to configure ACLs. ACLs are basically a type of QoS policy, and the commands used to configure ACLs are a subset of the switch's QoS commands. For information about basic configuration of QoS policies, see [Chapter 34, "Configuring QoS."](#)

To configure an ACL, the following general steps are required:

- 1 Set the global disposition.** This step is described in ["Setting the Global Disposition"](#) on page 35-7.
- 2 Create a condition for the traffic to be filtered.** This step is described in ["Creating Condition Groups For ACLs"](#) on page 35-8 and ["Creating Policy Conditions For ACLs"](#) on page 35-9.
- 3 Create an action to accept or deny the traffic.** This step is described in ["Creating Policy Actions For ACLs"](#) on page 35-10.
- 4 Create a policy rule that combines the condition and the action.** This step is described in ["Creating Policy Rules for ACLs"](#) on page 35-10.

For a quick tutorial on how to configure ACLs, see ["Quick Steps for Creating ACLs"](#) on page 35-4.

Setting the Global Disposition

By default, flows that do not match any policies are accepted on the switch. You may configure the switch to deny a bridged or multicast flow that does not match a policy.

Note. Note that the global disposition setting applies to all policy rules on the switch, not just those that are configured for ACLs.

The global commands include:

```
qos default bridged disposition
qos default multicast disposition
```

To change the global default dispositions, use these commands with the desired disposition value (**accept**, **drop**, or **deny**).

Note that in the current release of Alcatel-Lucent's QoS software, the **drop** and **deny** keywords produce the same result (flows are silently dropped; no ICMP message is sent).

The default disposition for routed flows is not configurable on a global basis for the switch. Policies may be set up to allow or deny any routed traffic through the switch.

For more information about the global disposition commands, see [Chapter 34, "Configuring QoS."](#) and the *OmniSwitch 6250 CLI Reference Guide*.

Important. If you set the global bridged disposition (using the **qos default bridged disposition** command) to **deny** or **drop**, it will result in dropping all Layer 2 traffic from the switch that does not match any policy to accept traffic. You must create policies (one for source and one for destination) to allow traffic on the switch.

If you set the bridged disposition to **deny** or **drop**, and you configure Layer 2 ACLs, you will need two rules for each type of filter. For more information, see [“Layer 2 ACLs” on page 35-10](#).

Creating Condition Groups For ACLs

Condition groups for ACLs are made up of multiple IP addresses (IPv4 only; IPv6 not supported with condition groups), MAC addresses, services, IP ports or VLANs to which you want to apply the same disposition. Instead of creating a separate condition for each policy rule, create a condition group and associate the group with the condition. This reduces the number of rules you would have to configure (one for each address, service, or port). The commands used for creating condition groups include:

- policy network group**
- policy mac group**
- policy service**
- policy service group**
- policy port group**
- policy vlan group**

For example:

```
-> policy network group netgroup2 10.10.5.1 10.10.5.2 10.10.5.3
-> policy condition cond2 source network group netgroup2
```

This command configures a network group (**netgroup2**) of three IP addresses. The network group is then configured as part of a policy condition (**cond2**). The condition specifies that the addresses in the group are source addresses. (For all condition groups except service groups, the policy condition specifies whether the condition group is a *source* or *destination* group.)

If a network group was not used, a separate condition would have to be created for each IP address. Subsequently, a corresponding rule would have to be created for each condition. Using a network group reduces the number of rules required.

For more details about using groups in policy conditions, see [“Using Condition Groups in Policies” on page 34-46 in Chapter 34, “Configuring QoS.”](#)

Configuring ACLs

This section describes in detail the procedures for configuring ACLs. For more information about how to configure policies in general, see [Chapter 34, “Configuring QoS.”](#) Command syntax is described in detail in the *OmniSwitch 6250 CLI Reference Guide*.

The basic commands for configuring ACL rules are the same as those for configuring policy rules:

- policy condition**
- policy action**
- policy rule**

Creating Policy Conditions For ACLs

A policy condition for IP filtering may include a particular source IP address, destination IP address, source IP port, or destination IP port. Or, the condition may simply refer to the network group, MAC group, port group, or service group. Typically ACLs use group keywords in policy conditions. A single rule, therefore, filters traffic for multiple addresses or ports.

For example:

```
-> policy port group pgroup1 3/1-2 4/3 5/4
-> policy condition c2 source port group pgroup1
```

In this example, a Layer 2 condition (**c2**) specifies that traffic matches the ports included of the **pgroup1** port group. The condition also specifies that the port group is a source group. Any traffic coming in on ports 1 or 2 on slot 3, port 3 on slot 4, or port 4 on slot 5 will match condition **c2**.

For more information about condition groups, see [“Creating Condition Groups For ACLs” on page 35-8](#).

The following table lists the keywords for the **policy condition** command that are typically used for the different types of ACLs:

Layer 2 ACL Condition Keywords	Layer 3/4 ACL Condition Keywords	Multicast ACL Condition Keywords
source mac	source ip	multicast ip
source mac group	source ipv6	multicast ipv6
destination mac	source network group	multicast network group
destination mac group	destination ip	destination ip
source vlan	destination ipv6	destination vlan
source vlan group	destination network group	destination port
source port	source ip port	destination port group
source port group	destination ip port	destination mac
ethertype	service	destination mac group
802.1p	service group	
	ip protocol	
	ipv6	
	icmptype	
	icmpcode	
	tos	
	dscp	
	source tcp port	
	destination tcp port	
	source udp port	
	destination udp port	
	established	
	tcpflags	

Note that the individual address, service, or port cannot be used in conjunction with the same type of condition group. For example, you cannot specify in the same rule both a source MAC address and a source MAC group.

Creating Policy Actions For ACLs

A policy action for IP filtering specifies a *disposition*, that is, whether the flow is accepted or denied on the switch. To create a policy action, use the **policy action** command. Use the **disposition** keyword to define whether the flow is accepted (**accept**) or denied (**deny**). For example:

```
-> policy action a1 disposition accept
```

If you do not specify a disposition for the policy action, the default (**accept**) will be used.

Creating Policy Rules for ACLs

A policy rule is made up of a condition and an action. For example, to create a policy rule for filtering IP addresses, which is a Layer 3 ACL, use the **policy rule** command with the **condition** and **action** keywords. The **precedence** keyword is optional. By default rules have a precedence of 0. See [“Rule Precedence” on page 35-6](#) for more information about precedence.

```
-> policy condition c3 source ip 10.10.4.8
-> policy action a1 accept
-> policy rule rule7 precedence 65535 condition c3 action a1
```

In this example, any traffic matching condition **c3** will match **rule7**; **rule7** is configured with the highest precedence value. If any other rules are configured for traffic with a source address of 10.10.4.8, **rule7** will take precedence over the other rules only if one of the following is true:

- A conflict exists with another rule and **rule7** has a higher precedence.
- A conflict exists with another rule that has the same precedence value, but **rule7** was created first.

The action configured for the rule, **a1**, allows traffic from 10.10.4.8, so the flow will be accepted on the switch.

The rule will not be used to classify traffic or enforce the policy until the **qos apply** command is entered. For information about applying policy parameters, see [“Applying the Configuration” on page 34-59](#) in Chapter 34, “Configuring QoS.”

Layer 2 ACLs

Layer 2 filtering filters traffic at the MAC layer. Layer 2 filtering may be done for both bridged and routed packets. As MAC addresses are learned on the switch, QoS classifies the traffic based on:

- MAC address or MAC group
- Source VLAN
- Physical slot/port or port group

The switch classifies the MAC address as both source *and* destination.

The following **policy condition** keywords are used for Layer 2 ACLs:

Layer 2 ACL Condition Keywords

source mac	802.1p
source mac group	destination mac
source vlan	destination mac group
source vlan group	destination vlan (multicast only)
source port	destination port (multicast only)
source port group	destination port group (multicast only)
ethertype	

A group and an individual item cannot be specified in the same condition. For example, a source MAC address and a source MAC group cannot be specified in the same condition.

Note that combining Layer 2 and Layer 3 conditions in the same policy is supported. Refer to [“Condition Combinations” on page 34-7](#) and [“Action Combinations” on page 34-9](#) in Chapter 34, “Configuring QoS.”

Layer 2 ACL Example

In this example, the default bridged disposition is **accept** (the default). Since the default is **accept**, the **qos default bridged disposition** command would only need to be entered if the disposition had previously been set to **deny**. The command is shown here for completeness.

```
-> qos default bridged disposition accept
-> policy condition Address1 source mac 080020:112233 source vlan 5
-> policy action BlockTraffic disposition deny
-> policy rule FilterA condition Address1 action BlockTraffic
```

In this scenario, traffic with a source MAC address of 08:00:20:11:22:33 coming in on VLAN 5 would match condition **Address1**, which is a condition for a policy rule called **FilterA**. **FilterA** is then applied to the flow. Since **FilterA** has an action (**BlockTraffic**) that is set to deny traffic, the flow would be denied on the switch.

Note that although this example contains only Layer 2 conditions, it is possible to combine Layer 2 and Layer 3 conditions in the same policy.

Layer 3 ACLs

The QoS software in the switch filters routed and bridged traffic at Layer 3.

For Layer 3 filtering, the QoS software in the switch classifies traffic based on:

- Source IP address or source network group
- Destination IP address or destination network group
- IP protocol
- ICMP code
- ICMP type
- Source TCP/UDP port
- Destination TCP/UDP port or service or service group

The following **policy condition** keywords are used for Layer 3 ACLs:

Layer 3/4 ACL Condition Keywords

source ip	tos
source network group	dscp
destination ip	source tcp port
destination network group	destination tcp port
multicast ip	source udp port
multicast network group	destination udp port
ip protocol	service
source ip port	service group
destination ip port	established
icmptype	tcpflags
icmpcode	

Note that combining Layer 2 and Layer 3 conditions in the same policy is supported. Refer to “[Condition Combinations](#)” on page 34-7 and “[Action Combinations](#)” on page 34-9 in Chapter 34, “Configuring QoS.”

Layer 3 ACL: Example 1

In this example, the default routed disposition is **accept** (the default).

```
-> policy condition addr2 source ip 192.68.82.0 source ip port 23 ip protocol 6
-> policy action Block disposition deny
-> policy rule FilterL31 condition addr2 action Block
```

Traffic with a source IP address of 192.68.82.0, a source IP port of 23, using protocol 6, will match condition **addr2**, which is part of **FilterL31**. The action for the filter (**Block**) is set to deny traffic. The flow will be dropped on the switch.

Note that although this example contains only Layer 2 conditions, it is possible to combine Layer 2 and Layer 3 conditions in the same policy.

Layer 3 ACL: Example 2

This example uses condition groups to combine multiple IP addresses in a single condition.

```
-> policy network group GroupA 192.60.22.1 192.60.22.2 192.60.22.0
-> policy condition cond7 destination network group GroupA
-> policy action Ok disposition accept
-> policy rule FilterL32 condition cond7 action Ok
```

In this example, a network group, **GroupA**, is configured with three IP addresses. Condition **cond7** includes **GroupA** as a destination group. Flows coming into the switch destined for any of the specified IP addresses in the group will match rule **FilterL32**. **FilterL32** is configured with an action (**Ok**) to allow the traffic on the switch.

Note that although this example contains only Layer 2 conditions, it is possible to combine Layer 2 and Layer 3 conditions in the same policy.

IPv6 ACLs

An ACL is considered an IPv6 ACL if the **ipv6** keyword and/or any of the following specific policy condition keywords are used in the ACL to classify/filter IPv6 traffic:

IPv6 ACL Keywords

source ipv6
destination ipv6
source tcp port
destination port (multicast only)
source udp port
destination udp port
ipv6

Note that IPv6 ACLs are effected only on IPv6 traffic. All other ACLs/policies with IP conditions that do not use the IPv6 keyword are effected only on IPv4 traffic. For example:

```
-> policy condition c1 tos 7
-> policy condition c2 tos 7 ipv6
```

In the above example, c1 is an IPv4 condition and c2 is an IPv6 condition. ACLs that use c1 are considered IPv4 policies; ACLs that use c2 are considered IPv6 policies. In addition, consider the following examples:

```
-> policy condition c3 source port 1/10
-> policy condition c4 source port 1/10 ipv6
```

Condition c3 applies to all traffic ingressing on port 1/10. However, condition c4 applies only to IPv6 traffic ingressing on port 1/10.

Note the following when configuring IPv6 ACLs:

- Trusted/untrusted behavior is the same for IPv6 traffic as it is for IPv4 traffic.
- IPv6 policies do not support the use of network groups, service groups, map groups, or MAC groups.
- IPv6 multicast policies are not supported.
- IPv6 policies are not supported by egress policy conditions.
- Anti-spoofing and other UserPorts profiles/filters do not support IPv6.
- The default (built-in) network group, “Switch”, only applies to IPv4 interfaces. There is no such group for IPv6 interfaces.

For more information regarding IPv6 condition parameters, see the [policy condition](#) command in the *OmniSwitch 6250 CLI Reference Guide*.

Multicast Filtering ACLs

Multicast filtering may be set up to filter clients requesting group membership via the Internet Group Management Protocol (IGMP). IGMP is used to track multicast group membership. The IP Multicast Switching (IPMS) function in the switch optimizes the delivery of IP multicast traffic by sending packets only to those stations that request it. Potential multicast group members may be filtered out so that IPMS does not send multicast packets to those stations.

For more information about IPMS, see [Chapter 36, “Configuring IP Multicast Switching.”](#)

Multicast traffic has its own global disposition. By default, the global disposition is **accept**. To change the default, use the **qos default multicast disposition** command.

For multicast filtering, the switch classifies traffic based on the multicast IP address or multicast network group and any destination parameters. Note that the destination parameters are used for the client from which the switch will receive the IGMP request.

The **multicast ip** or **multicast network group** keyword is required in the condition configured for a multicast ACL.

The following keywords may be used in the condition to indicate the client parameters:

Multicast ACL Keywords

destination ip
destination vlan
destination port
destination port group
destination mac
destination mac group

If a destination group is specified, the corresponding single value keyword cannot be combined in the same condition. For example, if a destination port is specified, a destination port group cannot be specified in the same condition.

To filter multicast clients, specify the multicast IP address, which is the address of the multicast group or stream, and specify the client IP address, VLAN, MAC address, or slot/port. For example:

```
-> qos default multicast disposition deny
-> policy condition Mclient1 multicast ip 224.0.1.2 destination vlan 5
-> policy action ok disposition accept
-> policy rule Mrule condition Mclient1 action ok
```

In this example, any traffic coming in on VLAN 5 requesting membership to the 224.0.1.2 multicast group will be allowed.

Using ACL Security Features

The following additional ACL features are available for improving network security and preventing malicious activity on the network:

- **UserPorts**—A port group that identifies its members as user ports to prevent source address spoofing of IP and ARP traffic (per RFC 2267). When a port is configured as a member of this group, packets received on the port are dropped if they contain a source IP address that does not match the IP subnet for the port. It is also possible to configure a UserPorts profile to specify other types of traffic to monitor on user ports. See [“Configuring a UserPorts Group” on page 35-15](#).
- **DropServices**—A service group that improves the performance of ACLs that are intended to deny packets destined for specific TCP/UDP ports. This group only applies to ports that are members of the UserPorts group. Using the DropServices group for this function minimizes processing overhead, which otherwise could lead to a DoS condition for other applications trying to use the switch. See [“Configuring a DropServices Group” on page 35-16](#).
- **ICMP drop rules**—Allows condition combinations in policies that will prevent user pings, thus reducing DoS exposure from pings. Two condition parameters are also available to provide more granular filtering of ICMP packets: **icmptype** and **icmrcode**. See [“Configuring ICMP Drop Rules” on page 35-17](#).
- **TCP connection rules**—Allows the determination of an *established* TCP connection by examining TCP flags found in the TCP header of the packet. Two condition parameters are available for defining a TCP connection ACL: **established** and **tcpflags**. See [“Configuring TCP Connection Rules” on page 35-17](#).
- **Early ARP discard**—ARP packets destined for other hosts are discarded to reduce processing overhead and exposure to ARP DoS attacks. No configuration is required to use this feature, it is always available and active on the switch. Note that ARPs intended for use by a local subnet and Local Proxy ARP are *not* discarded.
- **ARP ACLs**—It is also possible to create an ACL that will examine the source IP address in the header of ARP packets. This is done by specifying the ARP ethertype (0x0806) and source IP address.

Configuring a UserPorts Group

To prevent IP address spoofing and/or other types of traffic on specific ports, create a port group called **UserPorts** and add the ports to that group. For example, the following **policy port group** command adds ports 1/1-24, 2/1-24, 3/1, and 4/1 to the **UserPorts** group:

```
-> policy port group UserPorts 1/1-24 2/1-24 3/1 4/1
-> qos apply
```

Note that the UserPorts group applies to both bridged and routed traffic, and it is *not* necessary to include the UserPorts group in a condition and/or rule for the group to take effect. Once ports are designated as members of this group, IP spoofed traffic is blocked while normal traffic is still allowed on the port.

The UserPorts group is also used in conjunction with the DropServices group. If a flow received on a port that is a member of the UserPorts group is destined for a TCP or UDP port (service) specified in the DropServices group, the flow is dropped. See [“Configuring a DropServices Group” on page 35-16](#) for more information.

Configuring UserPort Traffic Types and Port Behavior

In addition to spoofed traffic, it is also possible to configure QoS to look for BPDU, RIP, and/or DHCP server packets on user ports. When the specified type of traffic is encountered, the user port can either filter the traffic or administratively shutdown to block all traffic.

By default spoofed traffic is filtered on user ports. To specify additional types of traffic to look for on these ports and select how the port will deal with such traffic, use the **qos user-port** command to configure a UserPorts profile. For example, the following command specifies that user ports should filter BPDU packets:

```
-> qos user-port filter spoof
```

To specify multiple types of traffic on the same command line, enter each type separated by a space. For example:

```
-> qos user-port filter bdpu rip
```

Note that a slot and port is not required with the **qos user-port** command. This is because the command applies to all ports that are members of the UserPorts group.

The following **qos user-port** command example uses the **shutdown** option to administratively disable the user port if the specified type of traffic is received on that port:

```
-> qos user-port shutdown bdpu
```

Note that an SNMP trap is sent whenever a user port shutdown occurs. To enable a port disabled by a user port shutdown operation, use the **interfaces admin** command to administratively enable the port or disconnect and reconnect the port cable.

To disable the filter or shutdown function, use the **no** form of the **qos user-port** command. For example, the following command disables the filtering operation for all user ports:

```
-> qos no user-port filter
```

Note that any changes to the UserPorts profile (e.g., adding or removing a traffic type) are not made until the **qos apply** command is performed.

Configuring a DropServices Group

To drop packets destined for specific TCP and UDP ports using minimal switch resources, configure a services group called **DropServices** with a list of previously defined TCP/UDP services. The DropServices group is used in conjunction with the UserPorts group. TCP/UDP services that belong to the DropServices group are only filtered on ports that belong to the UserPorts group.

Note that it is not necessary to include the DropServices group in an ACL for the group to take effect. DropServices is a reserved group that is active once TCP/UDP services are added to the group and ports are added to the reserved UserPorts group and the QoS configuration is applied. For example:

1 Create destination port services for the TCP/UDP traffic that you want dropped using the **policy service** command, as shown below:

```
-> policy service tcp135 destination tcp port 135
-> policy service tcp445 destination tcp port 445
-> policy service udp137 destination udp port 137
-> policy service udp138 destination udp port 138
-> policy service udp445 destination udp port 445
```

- 2 Add the services created in Step 1 to a service group called **DropServices** using the **policy service group** command, as shown below:

```
-> policy service group DropServices tcp135 tcp445 udp137 udp138 udp445
```

Note that the DropServices group must be specified using the exact capitalization as shown in the above example.

- 3 Add ports to the port group called **UserPorts** using the **policy port group** command, as shown below:

```
-> policy port group UserPorts 1/1 3/1-24
```

Note that the UserPorts group must be specified using the exact capitalization as shown in the above example.

- 4 Apply the QoS configuration using the **qos apply** command.

```
-> qos apply
```

When the above steps are performed, an implicit ACL is created on the switch that applies to all VLANs. This internal ACL takes precedence over any other policies configured on the switch.

Configuring ICMP Drop Rules

Combining a Layer 2 condition for source VLAN with a Layer 3 condition for IP protocol is supported. In addition, two new condition parameters are available to provide more granular filtering of ICMP packets: **icmptype** and **icmpcode**. Use these two conditions together in a policy to block ICMP echo request and reply packets without impacting switch performance.

The following example defines an ACL policy that prevents users from pinging by dropping echo request ICMP packets at the source port:

```
-> policy condition pingEchoRequest source vlan 10 icmptype 8
-> policy action drop disposition drop
-> policy rule noping10 condition pingEchoRequest action drop
-> qos apply
```

Note that the above policy only blocks ICMP echo traffic, all other ICMP traffic is still allowed.

Configuring TCP Connection Rules

Two condition parameters are available for defining a TCP connection ACL policy: **established** and **tcpflags**. An ACL can be defined using the **established** parameter to identify packets that are part of an established TCP connection and allow forwarding of the packets to continue. When this parameter is invoked, TCP header information is examined to determine if the **ack** or **rst** flag bit is set. If this condition is true, then the connection is considered established.

The following is an example ACL policy using the **established** condition parameter:

```
policy condition c destination ip 192.168.10.0 mask 255.255.255.0 established
policy condition c1 destination ip 192.168.10.0 mask 255.255.255.0
policy action drop disposition drop
policy action allow

policy rule r condition c action allow
policy rule r1 condition c1 action drop
qos apply
```

This example ACL policy will prevent any TCP connection from being initiated to the 192.168.10.0 network and all other IP traffic to the 192.168.10.0 network. Only TCP connections initiated from the 192.168.10.0 network are allowed.

Note that the above example ACL would prevent FTP sessions. See the [policy condition established](#) command page in the *OmniSwitch 6250 CLI Reference Guide* for more information.

An ACL can also be defined using the **tcpflags** parameter to examine and qualify specific TCP flags individually or in combination with other flags. This parameter can be used to prevent specific DOS attacks, such as the *christmas tree*.

The following example use the **tcpflags** condition parameter to determine if the F (fin) and S (syn) TCP flag bits are set to one and the A (ack) bit is set to zero:

```
-> policy condition c1 tcpflags all f s mask f s a
```

In this example, a match must occur on all the flags or the packet is not allowed. If the optional command keyword **any** was used, then a match need only occur on any one of the flags. For example, the following condition specifies that either the A (ack) bit or the R (rst) bit must equal one:

```
-> policy condition c1 tcpflags any a r mask a r
```

Note that if a flag is specified on the command line after the **any** or **all** keyword, then the match value is one. If the flag only appears as part of the **mask**, then the match value is zero. See the [policy condition tcpflags](#) command page in the *OmniSwitch 6250 CLI Reference Guide* for more information.

Verifying the ACL Configuration

To display information about ACLs, use the same **show** commands that are used for displaying any QoS policies. These commands include:

show policy list	Displays information about all pending and applied policy conditions or a particular policy condition configured on the switch. Use the applied keyword to display information about applied conditions only.
show policy action	Displays information about all pending and applied policy actions or a particular policy action configured on the switch. Use the applied keyword to display information about applied actions only.
show active policy rule meter-statistics	Displays information about all pending and applied policy rules or a particular policy rule.
show active policy list	Displays the pending and applied policy rules that are active (enabled) on the switch.
show qos config	Displays global QoS configuration parameters.

When a **show** command is used to display output for all pending and applied policy configuration, the following characters may appear in the display:

character	definition
+	Indicates that the policy rule has been modified or has been created since the last qos apply .
-	Indicates the policy object is pending deletion.
#	Indicates that the policy object differs between the pending/applied objects.

The following example shows all policy rules configured on the switch:

```
-> show policy rule
      Policy          From Prec  Enab  Act  Refl  Log  Trap  Save
my_rule          cli  0    Yes  Yes   No   No   Yes  Yes
Cnd/Act:         cond5 -> action2

+my_rule5        cli  0    Yes  No   No   No   Yes  Yes
Cnd/Act:         cond2 -> pri2

mac1             cli  0    Yes  No   No   No   Yes  Yes
Cnd/Act:         dmacl -> pri2
```

The display indicates that **my_rule** is active and is used to classify traffic on the switch (the Act field displays **Yes**). The rule **my_rule5** has been configured since the last **qos apply** command was entered, as indicated by the plus (+) sign. The rule will not be used to classify traffic until the next **qos apply**. The rule **mac1** is not active, as indicated by the **No** in the Act field.

To display only policy rules that are active (enabled) on the switch, use the **show active policy rule** command, as shown in the following example:

```
-> show active policy rule
```

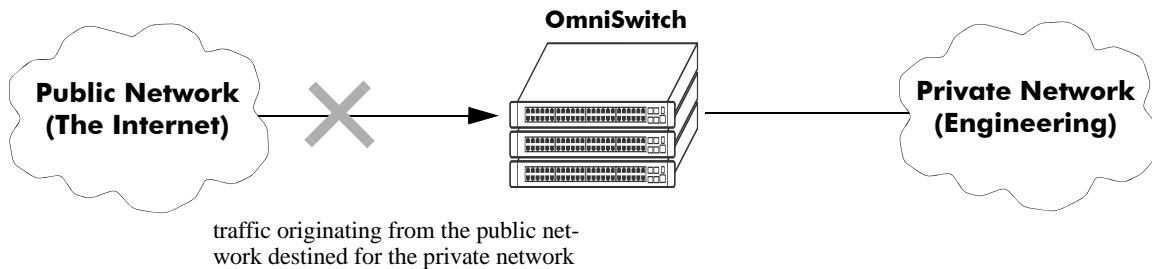
	Policy	From	Prec	Enab	Inact	Refl	Log	Save	Matches
+my_rule5		cli	0	Yes	No	No	No	Yes	0
Cnd/Act:		cond2	->	pri2					
mac1		cli	0	Yes	No	No	No	Yes	0
Cnd/Act:		dmac1	->	pri2					

In this example, the rule **my_rule** does not display because it is inactive. Rules are inactive if they are administratively disabled through the **policy rule** command, or if the rule cannot be enforced by the current hardware. Both **my_rule5** and **mac1** are displayed here because they are active; however, **my_rule5** is a pending rule and will not be used to classify traffic until the **qos apply** command is entered.

See the *OmniSwitch 6250 CLI Reference Guide* for more information about the output of these commands.

ACL Application Example

In this application for IP filtering, a policy is created to deny Telnet traffic from the outside world to an engineering group in a private network.



IP Filtering Application Example

Set up a policy rule called **outside** to deny Telnet traffic to the private network.

- 1 Create a policy service (**traffic_in**) for traffic originating from the well-known Telnet port number 23.

```
-> policy service traffic_in destination ip port 23 protocol 6
```

- 2 Create a policy condition (**outside_cond**) that references the service.

```
-> policy condition outside_cond service traffic_in
```

- 3 Create a policy action (**outside_action**) to deny the traffic.

```
-> policy action outside_action disposition drop
```

- 4 Then combine the condition and the action in a policy rule (**outside**).

```
-> policy rule outside condition outside_cond action outside_action
```

An example of what these commands look like together on consecutive command lines:

```
-> policy service traffic_in source ip port 23 protocol 6
-> policy condition outside_cond service traffic_in
-> policy action outside_action disposition drop
-> policy rule outside condition outside_cond action outside_action
```


36 Configuring IP Multicast Switching

IP Multicast Switching is a one-to-many communication technique employed by emerging applications, such as video distribution, news feeds, conferencing, netcasting, and resource discovery (RIP2 and BOOTP). Unlike unicast, which sends one packet per destination, multicast sends one packet to all devices in any subnetwork that has at least one device requesting the multicast traffic. Multicast switching also requires much less bandwidth than unicast techniques and broadcast techniques, since the source hosts only send one data stream to the ports on which destination hosts that request it are attached.

Destination hosts signal their intent to receive a specific IP multicast stream by sending a request to do so to a nearby switch by using Internet Group Management Protocol (IGMP). This is referred to as IGMP Snooping. Destination hosts signal their intent to receive a specific IPv6 multicast stream by sending a request to do so to a nearby switch by using Multicast listener discovery protocol (MLD). This is referred to as MLD Snooping. The switch then learns on which ports multicast group subscribers are attached and can intelligently deliver traffic only to the respective ports. Alcatel-Lucent's implementation of IGMP snooping is called IP Multicast Switching (IPMS) and MLD snooping is called IP Multicast Switching version 6 (IPMSv6). IPMS/IPMSv6 allows switches to efficiently deliver multicast traffic in hardware at wire speed.

In This Chapter

This chapter describes the basic components of IPMS and how to configure them through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling and disabling IP Multicast Switching on [page 36-8](#).
- Configuring and removing an IGMP static neighbor on [page 36-10](#).
- Configuring and removing an IGMP static querier on [page 36-11](#).
- Configuring and removing an IGMP static group on [page 36-11](#).
- Modifying IPMS parameters beginning on [page 36-13](#).
- Enabling and disabling IPv6 Multicast Switching on [page 36-22](#).
- Configuring and removing an MLD static neighbor on [page 36-24](#).
- Configuring and removing an MLD static querier on [page 36-25](#).
- Configuring and removing an MLD static group on [page 36-25](#).
- Modifying IPMSv6 parameters beginning on [page 36-27](#).

Note. You can also configure and monitor IPMS with WebView, Alcatel-Lucent's embedded Web-based device management application. WebView is an interactive and easy-to-use GUI that can be launched from OmniVista or a Web browser. Please refer to WebView's online documentation for more information on configuring and monitoring IPMS/IPMSv6 with WebView.

IPMS Specifications

The table below lists specifications for Alcatel-Lucent's IPMS software.

RFCs Supported	RFC 1112 — Host Extensions for IP Multicasting RFC 2236 — Internet Group Management Protocol, Version 2 RFC 2933 — Internet Group Management Protocol MIB RFC 3376 — Internet Group Management Protocol, Version 3
IETF Internet-Drafts Supported	draft-ietf-magma-snoop — Considerations for IGMP and MLD Snooping Switches
Platforms Supported	OmniSwitch 6250 Series
IGMP Versions Supported	IGMPv1, IGMPv2, IGMPv3
IGMP Query Interval	1 to 65535 in seconds
IGMP Router Timeout	1 to 65535 in seconds
IGMP Source Timeout	1 to 65535 in seconds
IGMP Query Response Interval	1 to 65535 in tenths of seconds
IGMP Last Member Query Interval	1 to 65535 in tenths of seconds

IPMSv6 Specifications

The table below lists specifications for Alcatel-Lucent's IPMSv6 software.

RFCs Supported	RFC 2710 — Multicast Listener Discovery for IPv6 RFC 3019 — IPv6 MIB for Multicast Listener Discovery Protocol RFC 3810 — Multicast Listener Discovery Version 2 for IPv6
IETF Internet-Drafts Supported	draft-ietf-magma-snoop — Considerations for IGMP and MLD Snooping Switches
Platforms Supported	OmniSwitch 6250 Series
MLD Versions Supported	MLDv1, MLDv2
MLD Query Interval	1 to 65535 in seconds
MLD Router Timeout	1 to 65535 in seconds
MLD Source Timeout	1 to 65535 in seconds
MLD Query Response Interval	1 to 65535 in milliseconds
MLD Last Member Query Interval	1 to 65535 in milliseconds

IPMS Default Values

The table below lists default values for Alcatel-Lucent's IPMS software.

Parameter Description	Command	Default Value/Comments
Administrative Status	ip multicast status	disabled
IGMP Querier Forwarding	ip multicast querier-forwarding	disabled
IGMP Version	ip multicast version	version 2
IGMP Query Interval	ip multicast query-interval	125 seconds
IGMP Last Member Query Interval	ip multicast last-member-query-interval	10 tenths-of-seconds
IGMP Query Response Interval	ip multicast query-response-interval	100 tenths-of-seconds
IGMP Router Timeout	ip multicast router-timeout	90 seconds
Source Timeout	ip multicast source-timeout	30 seconds
IGMP Querying	ip multicast querying	disabled
IGMP Robustness	ip multicast robustness	2
IGMP Spoofing	ip multicast spoofing	disabled
IGMP Zapping	ip multicast zapping	disabled

IPMSv6 Default Values

The table below lists default values for Alcatel-Lucent's IPMSv6 software.

Parameter Description	Command	Default Value/Comments
Administrative Status	ipv6 multicast status	disabled
MLD Querier Forwarding	ipv6 multicast querier-forwarding	disabled
MLD Version	ipv6 multicast version	version 1
MLD Query Interval	ipv6 multicast query-interval	125 seconds
MLD Last Member Query Interval	ipv6 multicast last-member-query-interval	1000 milliseconds
MLD Query Response Interval	ipv6 multicast query-response-interval	10000 milliseconds
MLD Router Timeout	ipv6 multicast router-timeout	90 seconds
Source Timeout	ipv6 multicast source-timeout	30 seconds
MLD Querying	ipv6 multicast querying	disabled
MLD Robustness	ipv6 multicast robustness	2
MLD Spoofing	ipv6 multicast spoofing	disabled
MLD Zapping	ipv6 multicast zapping	disabled

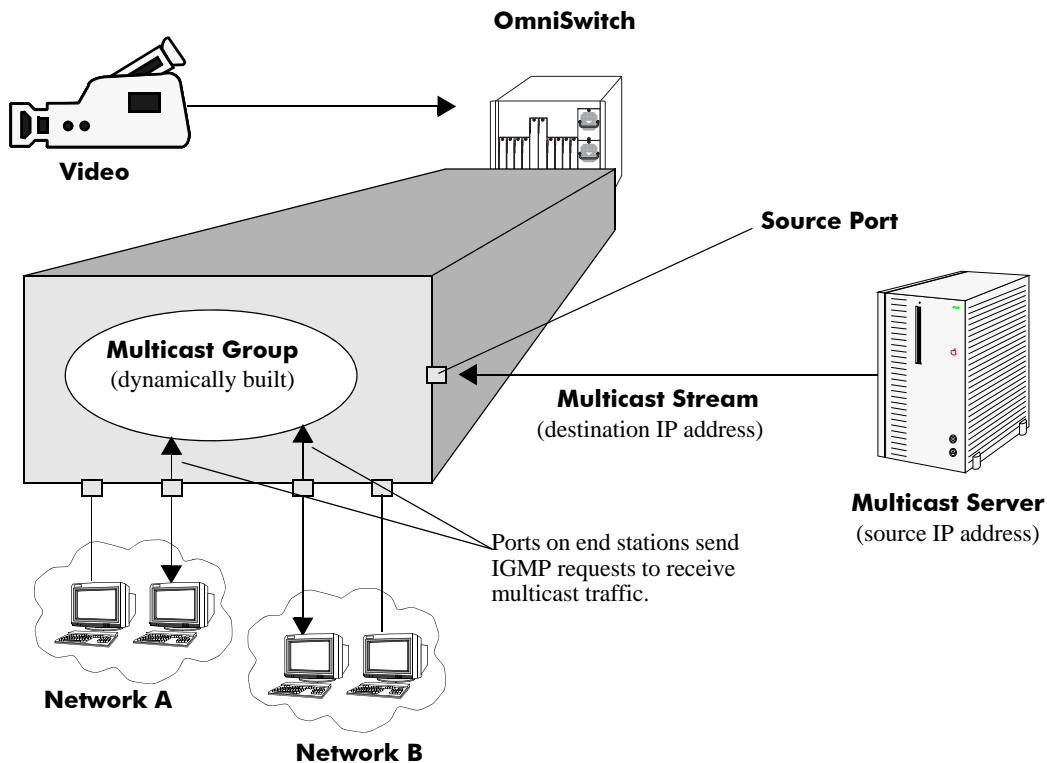
IPMS Overview

A multicast group is defined by a multicast group address, which is a Class D IP address in the range 224.0.0.0 to 239.255.255.255. (Addresses in the range 239.0.0.0 to 239.255.255.255 are reserved for boundaries.) The multicast group address is indicated in the destination address field of the IP header. (See [“Reserved IP Multicast Addresses” on page 36-7](#) for more information.)

IPMS tracks the source VLAN on which the Internet Group Management Protocol (IGMP) requests are received. The network interfaces verify that a multicast packet is received by the switch on the source (or expected) port.

IPMS Example

The figure on the following page shows an IPMS network where video content can be provided to clients that request it. A server is attached to the switch that provides the source (i.e., multicast) IP addresses. Clients from two different attached networks send IGMP reports to the switch to receive the video content.



Example of an IPMS Network

Reserved IP Multicast Addresses

The Internet Assigned Numbers Authority (IANA) created the range for multicast addresses, which is 224.0.0.0 to 239.255.255.255. However, as the table below shows, certain addresses are reserved and cannot be used.

Address or Address Range	Description
224.0.0.0 through 224.0.0.255	Routing protocols (e.g., RIP2)
224.0.1.0 through 224.0.1.255	Internetwork Control Block (e.g., RSVP, DHCP, commercial servers)
224.0.2.0 through 224.0.255.0	AD-HOC Block (e.g., commercial servers)
224.1.0.0 through 224.1.255.255	ST Multicast Groups
224.2.0.0 through 224.2.255.255	SDP/SAP Block
224.252.0.0 through 224.255.255.255	DIS Transient Groups
225.0.0.0 through 231.255.255.255	Reserved
232.0.0.0 through 232.255.255.255	Source Specific Multicast
233.0.0.0 through 233.255.255.255	GLOP Block
234.0.0.0 through 238.255.255.255	Reserved
239.0.0.0 through 239.255.255.255	Administratively Scoped

Configuring IPMS on a Switch

This section describes how to use Command Line Interface (CLI) commands to enable and disable IP Multicast Switching (IPMS) switch wide (see “[Enabling and Disabling IP Multicast Status](#)” on page 36-8), configure a port as a IGMP static neighbor (see “[Configuring and Removing an IGMP Static Neighbor](#)” on page 36-10), configure a port as a IGMP static querier (see “[Configuring and Removing an IGMP Static Querier](#)” on page 36-11), and configure a port as a IGMP static group (see “[Configuring and Removing an IGMP Static Group](#)” on page 36-11).

In addition, a tutorial is provided in “[IPMS Application Example](#)” on page 36-34 that shows how to use CLI commands to configure a sample network.

Note. See the “IP Multicast Switching Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of IPMS CLI commands.

Enabling and Disabling IP Multicast Status

IP Multicast Switching is disabled by default on a switch. The following subsections describe how to enable and disable IP Multicast Switching with the [ip multicast status](#) command.

Note. If IP Multicast switching is enabled on the system, the VLAN configuration overrides the system’s configuration.

Enabling IP Multicast Status

To enable IP Multicast switching on the system if no VLAN is specified, use the [ip multicast status](#) command as shown below:

```
-> ip multicast status enable
```

You can also enable IP Multicast switching on the specified VLAN by entering:

```
-> ip multicast vlan 2 status enable
```

Disabling IP Multicast Status

To disable IP Multicast switching on the system if no VLAN is specified, use the [ip multicast status](#) command as shown below:

```
-> ip multicast status disable
```

Or, as an alternative, enter:

```
-> ip multicast status
```

To restore the IP Multicast status to its default setting (i.e., disabled).

You can also disable IP Multicast switching on the specified VLAN by entering:

```
-> ip multicast vlan 2 status disable
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 status
```

To restore the IP Multicast status to its default setting (i.e., disabled).

Enabling and Disabling IGMP Querier-forwarding

By default, IGMP querier-forwarding is disabled. The following subsections describe how to enable and disable IGMP querier-forwarding by using the **ip multicast querier-forwarding** command.

Enabling the IGMP Querier-forwarding

You can enable the IGMP querier-forwarding by entering **ip multicast querier-forwarding** followed by the **enable** keyword. For example, to enable the IGMP querier-forwarding on the system if no VLAN is specified, you would enter:

```
-> ip multicast querier-forwarding enable
```

You can also enable the IGMP querier-forwarding on the specified VLAN by entering:

```
-> ip multicast vlan 2 querier-forwarding enable
```

Disabling the IGMP Querier-forwarding

You can disable the IGMP querier-forwarding by entering **ip multicast querier-forwarding** followed by the **disable** keyword. For example, to disable the IGMP querier-forwarding on the system if no VLAN is specified, you would enter:

```
-> ip multicast querier-forwarding disable
```

Or, as an alternative, enter:

```
-> ip multicast querier-forwarding
```

To restore the IGMP querier-forwarding to its default setting (i.e., disabled).

You can also disable the IGMP querier-forwarding on the specified VLAN by entering:

```
-> ip multicast vlan 2 querier-forwarding disable
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 querier-forwarding
```

To restore the IGMP querier-forwarding to its default setting (i.e., disabled).

You can remove an IGMP querier-forwarding entry on the specified VLAN and return to its default behavior by entering:

```
-> no ip multicast vlan 2 querier-forwarding
```

Configuring and Restoring the IGMP Version

By default, the version of Internet Group Management Protocol (IGMP) membership is Version 2. The following subsections describe how to configure IGMP protocol version ranging from 1 to 3 with the **ip multicast version** command.

Configuring the IGMP Version

To change the IGMP protocol version on the system if no VLAN is specified, use the **ip multicast version** command as shown below:

```
-> ip multicast version 3
```

You can also change the IGMP protocol version on the specified VLAN by entering:

```
-> ip multicast vlan 5 version 1
```

Restoring the IGMP Version

To restore the IGMP protocol version to its default (i.e., IGMPv2) version on the system if no VLAN is specified, use the **ip multicast version** command as shown below:

```
-> ip multicast version 0
```

Or, as an alternative, enter:

```
-> ip multicast version
```

To restore the IGMP version to its default version.

You can also restore the IGMP protocol version to version 2 on the specified VLAN by entering:

```
-> ip multicast vlan 2 version 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 version
```

To restore the IGMP version to its default version.

Configuring and Removing an IGMP Static Neighbor

IGMP static neighbor ports receive all multicast streams on the designated VLAN and also receive IGMP reports for the VLAN. The following subsections describe how to configure and remove a IGMP static neighbor port by using the **ip multicast static-neighbor** command.

Configuring an IGMP Static Neighbor

You can configure a port as an IGMP static neighbor port by entering **ip multicast static-neighbor** followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to configure port 10 in slot 4 with designated VLAN 2 as an IGMP static neighbor you would enter:

```
-> ip multicast static-neighbor vlan 2 port 4/10
```

You can also configure a link aggregation group as an IGMP static neighbor port by entering **ip multicast static-neighbor** followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, and the link aggregation group number.

For example, to configure link aggregation group 7 with designated VLAN 2 as a static neighbor you would enter:

```
-> ip multicast static-neighbor vlan 2 port 7
```

Removing an IGMP Static Neighbor

To reset the port so that it is no longer an IGMP static neighbor port, use the **no** form of the **ip multicast static-neighbor** command by entering **no ip multicast static-neighbor** followed by **vlan**, a space, VLAN number, a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to remove port 10 in slot 4 with designated VLAN 2 as an IGMP static neighbor you would enter:

```
-> no ip multicast static-neighbor vlan 2 port 4/10
```

Configuring and Removing an IGMP Static Querier

IGMP static querier ports receive IGMP reports generated on the designated VLAN. Unlike IPMS neighbor ports, they will not receive all multicast streams. The following subsections describe how to configure and remove a static querier by using the **ip multicast static-querier** command.

Configuring an IGMP Static Querier

You can configure a port as an IGMP static querier port by entering **ip multicast static-querier**, followed by **vlan**, a space, the VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to configure port 10 in slot 4 with designated VLAN 2 as an IGMP static querier you would enter:

```
-> ip multicast static-querier vlan 2 port 4/10
```

You can also configure a link aggregation group as an IGMP static querier port by entering **ip multicast static-querier** followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, and the link aggregation group number.

For example, to configure link aggregation group 7 with designated VLAN 2 as a static querier you would enter:

```
-> ip multicast static-querier vlan 2 port 7
```

Removing an IGMP Static Querier

To reset the port so that it is no longer an IGMP static querier port, use the **no** form of the **ip multicast static-querier** command by entering **no ip multicast static-querier**, followed by **vlan**, a space, the VLAN number, a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to remove port 10 in slot 4 with designated VLAN 2 as an IPMS static querier you would enter:

```
-> no ip multicast static-querier vlan 2 port 4/10
```

Configuring and Removing an IGMP Static Group

IGMP static group ports receive IGMP reports generated on the specified IP Multicast group address. The following subsections describe how to configure and remove a static group with the **ip multicast static-group** command.

Configuring an IGMP Static Group

You can configure a port as an IGMP static group by entering **ip multicast static-group**, followed by the IP address of the static group in dotted decimal notation, a space, followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to configure an IGMP static member with an IP address of 225.0.0.1 on port 10 in slot 3 with designated VLAN 3 you would enter:

```
-> ip multicast static-group 225.0.0.1 vlan 3 port 3/10
```

You can also configure a link aggregation group as an IPMS static group by entering **ip multicast static-group** followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, and the link aggregation group number.

For example, to configure link aggregation group 7 with designated VLAN 2 as a static group you would enter:

```
-> ip multicast static-group 225.0.0.2 vlan 2 port 7
```

Removing an IGMP Static Group

To reset the port so that it is no longer an IGMP static group port, use the **no** form of the **ip multicast static-group** command by entering **no ip multicast static-group**, followed by the IP address of the static group, a space, followed by **vlan**, a space, the VLAN number, a space, followed by **port**, the slot number of the port, a slash (/), and the port number.

For example, to remove an IGMP static member with an IP address of 225.0.0.1 on port 10 in slot 3 with designated VLAN 3 you would enter:

```
-> no ip multicast static-group 225.0.0.1 vlan 3 port 3/10
```


Modifying IPMS Parameters

The table in “[IPMS Default Values](#)” on page 36-4 lists default values for IPMS parameters. The following sections describe how to use CLI commands to modify these parameters.

Modifying the IGMP Query Interval

The default IGMP query interval (i.e., the time between IGMP queries) is 125 in seconds. The following subsections describe how to configure a user-specified query interval value and restore it with the [ip multicast query-interval](#) command.

Configuring the IGMP Query Interval

You can modify the IGMP query interval from 1 to 65535 in seconds by entering [ip multicast query-interval](#) followed by the new value. For example, to set the query interval to 60 seconds on the system if no VLAN is specified, you would enter:

```
-> ip multicast query-interval 60
```

You can also modify the IGMP query interval on the specified VLAN by entering:

```
-> ip multicast vlan 2 query-interval 60
```

Restoring the IGMP Query Interval

To restore the IGMP query interval to its default (i.e., 125 seconds) value on the system if no VLAN is specified, use the [ip multicast query-interval](#) command by entering:

```
-> ip multicast query-interval 0
```

Or, as an alternative, enter:

```
-> ip multicast query-interval
```

To restore the IGMP query interval to its default value.

You can also restore the IGMP query interval to its default value on the specified VLAN by entering:

```
-> ip multicast vlan 2 query-interval 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 query-interval
```

To restore the IGMP query interval to its default value.

Modifying the IGMP Last Member Query Interval

The default IGMP last member query interval (i.e., the time period to reply to an IGMP query message sent in response to a leave group message) is 10 in tenths of seconds. The following subsections describe how to configure the IGMP last member query interval and restore it by using the [ip multicast last-member-query-interval](#) command.

Configuring the IGMP Last Member Query Interval

You can modify the IGMP last member query interval from 1 to 65535 in tenths of seconds by entering **ip multicast last-member-query-interval** followed by the new value. For example, to set the IGMP last member query interval to 60 tenths-of-seconds on the system if no VLAN is specified, you would enter:

```
-> ip multicast last-member-query-interval 60
```

You can also modify the IGMP last member query interval on the specified VLAN by entering:

```
-> ip multicast vlan 3 last-member-query-interval 60
```

Restoring the IGMP Last Member Query Interval

To restore the IGMP last member query interval to its default (i.e., 10 tenths-of-seconds) value on the system if no VLAN is specified, use the **ip multicast last-member-query-interval** command by entering:

```
-> ip multicast last-member-query-interval 0
```

Or, as an alternative, enter:

```
-> ip multicast last-member-query-interval
```

To restore the IGMP last member query interval to its default value.

You can also restore the IGMP last member query interval on the specified VLAN by entering:

```
-> ip multicast vlan 2 last-member-query-interval 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 last-member-query-interval
```

To restore the IGMP last member query interval to its default value.

Modifying the IGMP Query Response Interval

The default IGMP query response interval (i.e., the time period to reply to an IGMP query message) is 100 in tenths of seconds. The following subsections describe how to configure the query response interval and how to restore it with the **ip multicast query-response-interval** command.

Configuring the IGMP Query Response Interval

You can modify the IGMP query response interval from 1 to 65535 in tenths of seconds by entering **ip multicast query-response-interval** followed by the new value. For example, to set the IGMP query response interval to 6000 tenths-of-seconds you would enter:

```
-> ip multicast query-response-interval 6000
```

You can also modify the IGMP query response interval on the specified VLAN by entering:

```
-> ip multicast vlan 3 query-response-interval 6000
```

Restoring the IGMP Query Response Interval

To restore the IGMP query response interval to its default (i.e., 100 tenths-of-seconds) value on the system if no VLAN is specified, use the **ip multicast query-response-interval** command by entering:

```
-> ip multicast query-response-interval 0
```

Or, as an alternative, enter:

```
-> ip multicast query-response-interval
```

To restore the IGMP query response interval to its default value.

You can also restore the IGMP query response interval on the specified VLAN by entering:

```
-> ip multicast vlan 2 query-response-interval 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 query-response-interval
```

To restore the IGMP query response interval to its default value.

Modifying the IGMP Router Timeout

The default IGMP router timeout (i.e., expiry time of IP multicast routers) is 90 seconds. The following subsections describe how to configure a user-specified router timeout value and how to restore it with the **ip multicast router-timeout** command.

Configuring the IGMP Router Timeout

You can modify the IGMP router timeout from 1 to 65535 seconds by entering **ip multicast router-timeout** followed by the new value. For example, to set the IGMP router timeout to 360 seconds on the system if no VLAN is specified, you would enter:

```
-> ip multicast router-timeout 360
```

You can also modify the IGMP router timeout on the specified VLAN by entering:

```
-> ip multicast vlan 2 router-timeout 360
```

Restoring the IGMP Router Timeout

To restore the IGMP router timeout to its default (i.e., 90 seconds) value on the system if no VLAN is specified, use the **ip multicast router-timeout** command by entering:

```
-> ip multicast router-timeout 0
```

Or, as an alternative, enter:

```
-> ip multicast router-timeout
```

To restore the IGMP router timeout to its default value.

You can also restore the IGMP router timeout on the specified VLAN by entering:

```
-> ip multicast vlan 2 router-timeout 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 router-timeout
```

To restore the IGMP router timeout to its default value.

Modifying the Source Timeout

The default source timeout (i.e., the expiry time of IP multicast sources) is 30 seconds. The following subsections describe how to configure a user-specified source timeout value and restore it by using the [ip multicast router-timeout](#) command.

Configuring the Source Timeout

You can modify the source timeout from 1 to 65535 seconds by entering [ip multicast source-timeout](#) followed by the new value. For example, to set the source timeout to 360 seconds on the system if no VLAN is specified, you would enter:

```
-> ip multicast source-timeout 360
```

You can also modify the source timeout on the specified VLAN by entering:

```
-> ip multicast vlan 2 source-timeout 360
```

Restoring the Source Timeout

To restore the source timeout to its default (i.e., 30 seconds) value on the system if no VLAN is specified, use the [ip multicast source-timeout](#) command by entering:

```
-> ip multicast source-timeout 0
```

Or, as an alternative, enter:

```
-> ip multicast source-timeout
```

To restore the source timeout to its default value.

You can also restore the source timeout on the specified VLAN by entering:

```
-> ip multicast vlan 2 source-timeout 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 source-timeout
```

To restore the source timeout to its default value.

Enabling and Disabling IGMP Querying

By default, IGMP querying is disabled. The following subsections describe how to enable and disable IGMP querying by using the **ip multicast querying** command.

Enabling the IGMP Querying

You can enable the IGMP querying by entering **ip multicast querying** followed by the **enable** keyword. For example, to enable the IGMP querying on the system if no VLAN is specified, you would enter:

```
-> ip multicast querying enable
```

You can also enable the IGMP querying on the specified VLAN by entering:

```
-> ip multicast vlan 2 querying enable
```

Disabling the IGMP Querying

You can disable the IGMP querying by entering **ip multicast querying** followed by the **disable** keyword. For example, to disable the IGMP querying on the system if no VLAN is specified, you would enter:

```
-> ip multicast querying disable
```

Or, as an alternative, enter:

```
-> ip multicast querying
```

To restore the IGMP querying to its default setting (i.e., disabled).

You can also disable the IGMP querying on the specified VLAN by entering:

```
-> ip multicast vlan 2 querying disable
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 querying
```

To restore the IGMP querying to its default setting (i.e., disabled).

You can remove an IGMP querying entry on the specified VLAN and return to its default behavior by entering:

```
-> no ip multicast vlan 2 querying
```

Modifying the IGMP Robustness Variable

The default value of the IGMP robustness variable (i.e., the variable that allows fine-tuning on a network, where the expected packet loss is higher) is 2. The following subsections describe how to set the value of the robustness variable and restore it with the **ip multicast robustness** command.

Configuring the IGMP Robustness variable

You can modify the IGMP robustness variable from 1 to 7 on the system if no VLAN is specified, by entering **ip multicast robustness** followed by the new value. For example, to set the value of IGMP robustness to 3 you would enter:

```
-> ip multicast robustness 3
```

Note. If the links are known to be lossy, then robustness variable can be set to a higher value (7).

You can also modify the IGMP robustness variable from 1 to 7 on the specified VLAN by entering:

```
-> ip multicast vlan 2 robustness 3
```

Restoring the IGMP Robustness Variable

You can restore the IGMP robustness variable to its default (i.e., 2) value on the system if no vlan is specified, by entering **ip multicast robustness** followed by the value 0 as shown below:

```
-> ip multicast robustness 0
```

Or, as an alternative, enter:

```
-> ip multicast robustness
```

To restore the IGMP robustness to its default value.

You can also restore the IGMP robustness variable to its default (i.e., 2) value on the specified VLAN, by entering **ip multicast robustness** followed by the value 0 as shown below:

```
-> ip multicast vlan 2 robustness 0
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 robustness
```

To restore the IGMP robustness to its default value.

Enabling and Disabling the IGMP Spoofing

By default, IGMP spoofing (i.e., replacing a client's MAC and IP address with the system's MAC and IP address, when proxying aggregated IGMP group membership information) is disabled on the switch. The following subsections describe how to enable and disable spoofing by using the **ip multicast spoofing** command.

Enabling the IGMP Spoofing

To enable IGMP spoofing on the system if no VLAN is specified, use the **ip multicast spoofing** command as shown below:

```
-> ip multicast spoofing enable
```

You can also enable IGMP spoofing on the specified VLAN by entering:

```
-> ip multicast vlan 2 spoofing enable
```

Disabling the IGMP Spoofing

To disable IGMP spoofing on the system if no VLAN is specified, use the **ip multicast spoofing** command as shown below:

```
-> ip multicast spoofing disable
```

Or, as an alternative, enter:

```
-> ip multicast spoofing
```

To restore the IGMP spoofing to its default setting (i.e., disabled).

You can also disable IGMP spoofing on the specified VLAN by entering:

```
-> ip multicast vlan 2 spoofing disable
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 spoofing
```

To restore the IGMP spoofing to its default setting (i.e., disabled).

You can remove an IGMP spoofing entry on the specified VLAN and return to its default behavior by entering:

```
-> no ip multicast vlan 2 spoofing
```

Enabling and Disabling the IGMP Zapping

By default, IGMP zapping (i.e., processing membership and source filter removals immediately without waiting for the protocol's specified time period – this mode facilitates IP TV applications looking for quick changes between IP multicast groups) is disabled on a switch. The following subsections describe how to enable and disable IGMP zapping by using the **ip multicast zapping** command.

Enabling the IGMP Zapping

To enable IGMP zapping on the system if no VLAN is specified, use the **ip multicast zapping** command as shown below:

```
-> ip multicast zapping enable
```

You can also enable IGMP zapping on the specified VLAN by entering:

```
-> ip multicast vlan 2 zapping enable
```

Disabling the IGMP Zapping

To disable IGMP zapping on the system if no VLAN is specified, use the **ip multicast zapping** command as shown below:

```
-> ip multicast zapping disable
```

Or, as an alternative, enter:

```
-> ip multicast zapping
```

To restore the IGMP zapping to its default setting (i.e., disabled).

You can also disable IGMP zapping on the specified VLAN by entering:

```
-> ip multicast vlan 2 zapping disable
```

Or, as an alternative, enter:

```
-> ip multicast vlan 2 zapping
```

To restore the IGMP zapping to its default setting (i.e., disabled).

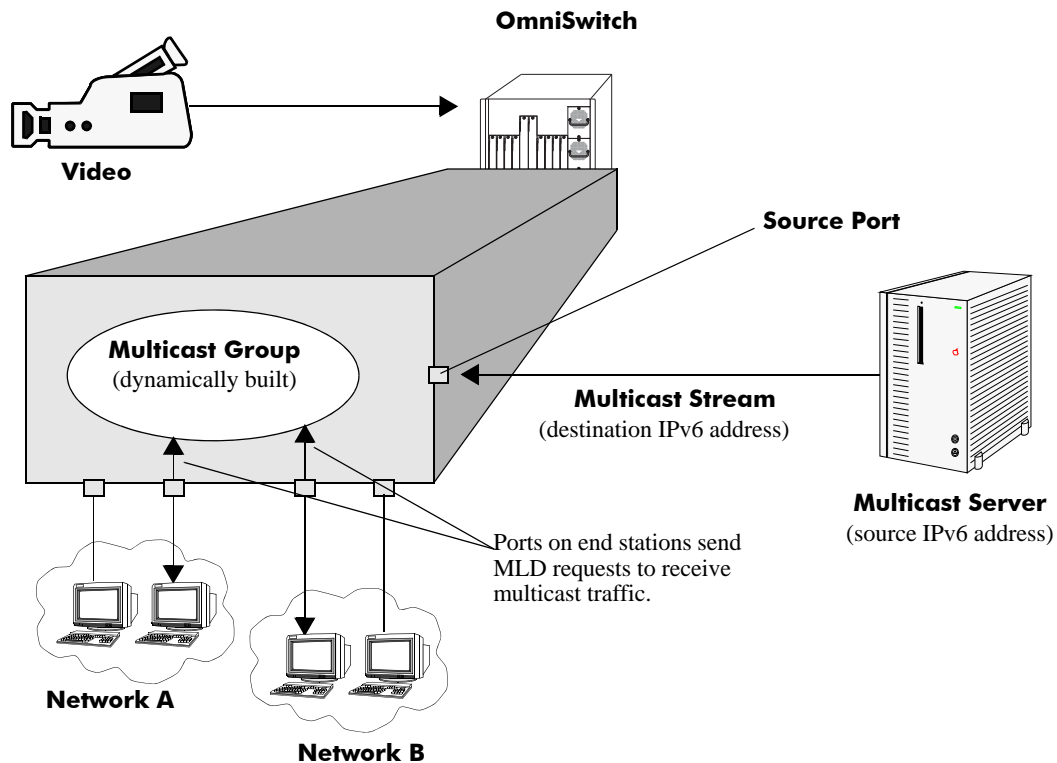
IPMSv6 Overview

An IPv6 multicast address identifies a group of nodes. A node can belong to any number of multicast groups. IPv6 multicast addresses are classified as fixed scope multicast addresses and variable scope multicast addresses. (See the “[Reserved IPv6 Multicast Addresses](#)” on page 36-21.)

IPMSv6 tracks the source VLAN on which the Multicast Listener Discovery Protocol (MLD) requests are received. The network interfaces verify that a multicast packet is received by the switch on the source (or expected) port.

IPMSv6 Example

The figure on the following page shows an IPMSv6 network where video content can be provided to clients that request it. A server is attached to the switch that provides the source (i.e., multicast) IPv6 addresses. Clients from two different attached networks send MLD reports to the switch to receive the video content.



Reserved IPv6 Multicast Addresses

The Internet Assigned Numbers Authority (IANA) classified the scope for IPv6 multicast addresses as fixed scope multicast addresses and variable scope multicast addresses. However, as the table below shows only well-known addresses, which are reserved and cannot be assigned to any multicast group.

Address	Description
FF00:0:0:0:0:0:0:0	reserved
FF01:0:0:0:0:0:0:0	node-local scope address
FF02:0:0:0:0:0:0:0	link-local scope
FF03:0:0:0:0:0:0:0	unassigned
FF04:0:0:0:0:0:0:0	unassigned
FF05:0:0:0:0:0:0:0	site-local scope
FF06:0:0:0:0:0:0:0	unassigned
FF07:0:0:0:0:0:0:0	unassigned
FF08:0:0:0:0:0:0:0	organization-local scope
FF09:0:0:0:0:0:0:0	unassigned
FF0A:0:0:0:0:0:0:0	unassigned
FF0B:0:0:0:0:0:0:0	unassigned
FF0C:0:0:0:0:0:0:0	unassigned
FF0D:0:0:0:0:0:0:0	unassigned
FF0E:0:0:0:0:0:0:0	global scope
FF0F:0:0:0:0:0:0:0	reserved

MLD Version 2

MLD is used by IPv6 systems (hosts and routers) to report their IPv6 multicast group memberships to any neighboring multicast routers. MLD Version 1 (MLDv1) handles forwarding by IPv6 multicast destination addresses only. MLD Version 2 (MLDv2) handles forwarding by source IPv6 addresses and IPv6 multicast destination addresses. Both MLDv1 and MLDv2 are supported.

Note. See [“Configuring the MLD Version 2”](#) on page 36-23 for information on configuring the IGMP version.

MLDv2 uses source filtering and reports multicast memberships to neighboring routers by sending membership reports. MLDv2 also supports Source Specific Multicast (SSM) by allowing hosts to report interest in receiving packets only from specific source addresses or from all but specific source addresses.

Configuring IPMSv6 on a Switch

This section describes how to use Command Line Interface (CLI) commands to enable and disable IPv6 Multicast Switching (IPMSv6) switch wide (see “[Enabling and Disabling IPv6 Multicast Status](#)” on page 36-22), configure a port as an MLD static neighbor (see “[Configuring and Removing an MLD Static Neighbor](#)” on page 36-24), configure a port as an MLD static querier (see “[Configuring and Removing an MLD Static Querier](#)” on page 36-25), and configure a port as an MLD static group (see “[Configuring and Removing an MLD Static Group](#)” on page 36-25)

Note. See the “IP Multicast Switching Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of IPMSv6 CLI commands.

Enabling and Disabling IPv6 Multicast Status

IPv6 Multicast is disabled by default on a switch. The following subsections describe how to enable and disable IPv6 Multicast by using the [ipv6 multicast status](#) command.

Note. If IPv6 Multicast switching is enabled on the system, the VLAN configuration overrides the system’s configuration.

Enabling IPv6 Multicast Status

To enable IPv6 Multicast switching on the system if no VLAN is specified, use the [ipv6 multicast status](#) command as shown below:

```
-> ipv6 multicast status enable
```

You can also enable IPv6 Multicast switching on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 status enable
```

Disabling IPv6 Multicast Status

To disable IPv6 Multicast switching on the system if no VLAN is specified, use the [ipv6 multicast status](#) command as shown below:

```
-> ipv6 multicast status disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast status
```

To restore the IPv6 Multicast status to its default setting.

You can also disable IPv6 Multicast on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 status disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 status
```

To restore the IPv6 Multicast status to its default setting.

Enabling and Disabling MLD Querier-forwarding

By default, MLD querier-forwarding is disabled. The following subsections describe how to enable and disable MLD querier-forwarding by using the **ipv6 multicast querier-forwarding** command.

Enabling the MLD Querier-forwarding

You can enable the MLD querier-forwarding by entering **ipv6 multicast querier-forwarding** followed by the **enable** keyword. For example, to enable the MLD querier-forwarding on the system if no VLAN is specified, you would enter:

```
-> ipv6 multicast querier-forwarding enable
```

You can also enable the MLD querier-forwarding on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 querier-forwarding enable
```

Disabling the MLD Querier-forwarding

You can disable the MLD querier-forwarding by entering **ipv6 multicast querier-forwarding** followed by the **disable** keyword. For example, to disable the MLD querier-forwarding on the system if no VLAN is specified, you would enter:

```
-> ipv6 multicast querier-forwarding disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast querier-forwarding
```

To restore the MLD querier-forwarding to its default setting (i.e., disabled).

You can also disable the MLD querier-forwarding on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 querier-forwarding disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 querier-forwarding
```

To restore the MLD querier-forwarding to its default setting (i.e., disabled).

You can remove an MLD querier-forwarding entry on the specified VLAN and return to its default behavior by entering:

```
-> no ipv6 multicast vlan 2 querier-forwarding
```

Configuring and Restoring the MLD Version

By default, the version of Multicast Listener Discovery (MLD) Protocol is Version 1. The following subsections describe how to configure the MLD version as Version 1 or Version 2 by using the **ipv6 multicast version** command.

Configuring the MLD Version 2

To change the MLD version to Version 2 (MLDv2) on the system if no VLAN is specified, use the **ipv6 multicast version** command as shown below:

```
-> ipv6 multicast version 2
```

Restoring the MLD Version 1

To restore the MLD version to Version 1 (MLDv1) on the system if no VLAN is specified, use the **ipv6 multicast version** command by entering:

```
-> ipv6 multicast version 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast version
```

To restore the MLD version to Version 1.

You can also restore the MLD version to Version 1 (MLDv1) on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 version 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 version
```

To restore the MLD version to Version 1.

Configuring and Removing an MLD Static Neighbor

MLD static neighbor ports receive all multicast streams on the designated VLAN and also receive MLD reports for the VLAN. The following subsections describe how to configure and remove a static neighbor port by using the **ipv6 multicast static-neighbor** command.

Configuring an MLD Static Neighbor

You can configure a port as an MLD static neighbor port by entering **ipv6 multicast static-neighbor** followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to configure port 10 in slot 4 with designated VLAN 2 as an MLD static neighbor you would enter:

```
-> ipv6 multicast static-neighbor vlan 2 port 4/10
```

You can also configure a link aggregation group as an MLD static neighbor port by entering **ipv6 multicast static-neighbor** followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, and the link aggregation group number.

For example, to configure link aggregation group 7 with designated VLAN 2 as a static neighbor you would enter:

```
-> ipv6 multicast static-neighbor vlan 2 port 7
```

Removing an MLD Static Neighbor

To reset the port so that it is no longer an MLD static neighbor port, use the **no** form of the **ipv6 multicast static-neighbor** command by entering **no ipv6 multicast static-neighbor**, followed by **vlan**, a space, the VLAN number, a space, followed by **port**, a space, slot number of the port, a slash (/), and the port number.

For example, to remove port 10 in slot 4 with designated VLAN 2 as an MLD static neighbor you would enter:

```
-> no ipv6 multicast static-neighbor vlan 2 port 4/10
```

Configuring and Removing an MLD Static Querier

MLD static querier ports receive MLD reports generated on the designated VLAN. Unlike MLD neighbor ports, they will not receive all multicast streams. The following subsections describe how to configure and remove a static querier by using the **ipv6 multicast static-querier** command.

Configuring an MLD Static Querier

You can configure a port as an MLD static querier port by entering **ipv6 multicast static-querier**, followed by **vlan**, a space, the VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to configure port 10 in slot 4 with designated VLAN 2 as an MLD static querier you would enter:

```
-> ipv6 multicast static-querier vlan 2 port 4/10
```

You can also configure a link aggregation group as an MLD static querier port by entering **ipv6 multicast static-querier**, followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, and the link aggregation group number.

For example, to configure link aggregation group 7 with designated VLAN 2 as a static querier you would enter:

```
-> ipv6 multicast static-querier vlan 2 port 7
```

Removing an MLD Static Querier

To reset the port, so that it is no longer an MLD static querier port, use the **no** form of the **ipv6 multicast static-querier** command by entering **no ipv6 multicast static-querier**, followed by **vlan**, a space, the VLAN number, a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to remove port 10 in slot 4 with designated VLAN 2 as a static querier you would enter:

```
-> no ipv6 multicast static-querier vlan 2 port 4/10
```

Configuring and Removing an MLD Static Group

MLD static group ports receive MLD reports generated on the specified IPv6 Multicast group address. The following subsections describe how to configure and remove an MLD static group by using the **ipv6 multicast static-group** command.

Configuring an MLD Static Group

You can configure a port as an MLD static group by entering **ipv6 multicast static-group**, followed by the IPv6 address of the MLD static group in hexadecimal notation separated by colons, a space, followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, the slot number of the port, a slash (/), and the port number.

For example, to configure an MLD static group with an IPv6 address of `ff05::5` enter:

```
-> ipv6 multicast static-group ff05::5 vlan 3 port 3/10
```

You can also configure a link aggregation group as an MLD static group by entering **ipv6 multicast static-group**, followed by **vlan**, a space, VLAN number (which must be between 0 and 4095), a space, followed by **port**, a space, and the link aggregation group number.

For example, to configure link aggregation group 7 with designated VLAN 2 as a static group you would enter:

```
-> ipv6 multicast static-group ff05::6 vlan 2 port 7
```

Removing an MLD Static Group

To reset the port so that it is no longer an MLD static group port, use the **no** form of the **ipv6 multicast static-group** command by entering **no ipv6 multicast static-group**, followed by the IPv6 address of the static group in hexadecimal notation separated by colons, a space, followed by **vlan**, a space, VLAN number, a space, followed by **port**, a space, the slot number of the port, a slash (/), and the port number.

For example, to remove an MLD static member with an IPv6 address of `ff05::5` on port 10 in slot 3 with designated VLAN 3 you would enter:

```
-> no ipv6 multicast static-group ff05::5 vlan 3 port 3/10
```

Modifying IPMSv6 Parameters

The table in “[IPMSv6 Default Values](#)” on page 36-5 lists default values for IPMSv6 parameters. The following sections describe how to use CLI commands to modify these parameters.

Modifying the MLD Query Interval

The default IPMSv6 query interval (i.e., the time between MLD queries) is 125 in seconds. The following subsections describe how to configure a user-specified query interval value and restore it by using the [ipv6 multicast query-interval](#) command.

Configuring the MLD Query Interval

You can modify the MLD query interval from 1 to 65535 in seconds by entering [ipv6 multicast query-interval](#) followed by the new value. For example, to set the MLD query interval to 60 seconds on the system if no VLAN is specified, you would enter:

```
-> ipv6 multicast query-interval 160
```

You can also modify the MLD query interval on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 query-interval 160
```

Restoring the MLD Query Interval

To restore the MLD query interval to its default (i.e., 125 seconds) value on the system if no VLAN is specified, use the [ipv6 multicast query-interval](#) command by entering:

```
-> no ipv6 multicast query-interval
```

You can also restore the MLD query interval on the specified VLAN by entering:

```
-> no ipv6 multicast vlan 2 query-interval
```

Modifying the MLD Last Member Query Interval

The default MLD last member query interval (i.e., the time period to reply to an MLD query message sent in response to a leave group message) is 1000 in milliseconds. The following subsections describe how to configure the MLD last member query interval and restore it by using the [ipv6 multicast last-member-query-interval](#) command.

Configuring the MLD Last Member Query Interval

You can modify the MLD last member query interval from 1 to 65535 in milliseconds by entering [ipv6 multicast last-member-query-interval](#) followed by the new value. For example, to set the MLD last member query interval to 600 milliseconds on the system if no VLAN is specified, you would enter:

```
-> ipv6 multicast last-member-query-interval 2200
```

You can also modify the MLD last member query interval on the specified VLAN by entering:

```
-> ipv6 multicast vlan 3 last-member-query-interval 2200
```

Restoring the MLD Last Member Query Interval

To restore the MLD last member query interval to its default (i.e., 1000 milliseconds) value on the system if no VLAN is specified, use the **ipv6 multicast last-member-query-interval** command by entering:

```
-> ipv6 multicast last-member-query-interval 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast last-member-query-interval
```

To restore the MLD last member query interval to its default (i.e., 1000 milliseconds) value.

You can also restore the MLD last member query interval on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 last-member-query-interval 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 last-member-query-interval
```

To restore the MLD last member query interval to its default (i.e., 1000 milliseconds) value.

Modifying the MLD Query Response Interval

The default MLD query response interval (i.e., the time period to reply to an MLD query message) is 10000 in milliseconds. The following subsections describe how to configure the MLD query response interval and restore it by using the **ipv6 multicast query-response-interval** command.

Configuring the MLD Query Response Interval

You can modify the MLD query response interval from 1 to 65535 in milliseconds by entering **ipv6 multicast last-member-query-interval** followed by the new value. For example, to set the MLD query response interval to 6000 milliseconds you would enter:

```
-> ipv6 multicast query-response-interval 20000
```

You can also modify the MLD query response interval on the specified VLAN by entering:

```
-> ipv6 multicast vlan 3 query-response-interval 20000
```

Restoring the MLD Query Response Interval

To restore the MLD query response interval to its default (i.e., 10000 milliseconds) value on the system if no VLAN is specified, use the **ipv6 multicast query-response-interval** command by entering:

```
-> ipv6 multicast query-response-interval 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast query-response-interval
```

To restore the MLD query response interval to its default value.

You can also restore the MLD query response interval on the specified VLAN by entering:

```
-> ipv6 multicast van 2 query-response-interval 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 query-response-interval
```

To restore the MLD query response interval to its default value.

Modifying the MLD Router Timeout

The default MLD router timeout (i.e., expiry time of IPv6 multicast routers) is 90 seconds. The following subsections describe how to configure a user-specified router timeout value and restore it by using the [ipv6 multicast router-timeout](#) command.

Configuring the MLD Router Timeout

You can modify the MLD router timeout from 1 to 65535 seconds by entering [ipv6 multicast router-timeout](#) followed by the new value. For example, to set the MLD router timeout to 360 seconds on the system if no VLAN is specified, you would enter:

```
-> ipv6 multicast router-timeout 360
```

You can also modify the MLD router timeout on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 router-timeout 360
```

Restoring the MLD Router Timeout

To restore the MLD router timeout to its default (i.e., 90 seconds) value on the system if no VLAN is specified, use the [ipv6 multicast router-timeout](#) command by entering:

```
-> ipv6 multicast router-timeout 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast router-timeout
```

To restore the MLD router timeout to its default value.

You can also restore the MLD router timeout on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 router-timeout 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 router-timeout
```

To restore the MLD router timeout to its default value.

Modifying the Source Timeout

The default source timeout (i.e., expiry time of IPv6 multicast sources) is 30 seconds. The following subsections describe how to configure a user-specified source timeout value and restore it by using the [ipv6 multicast source-timeout](#) command.

Configuring the Source Timeout

You can modify the source timeout from 1 to 65535 seconds by entering **ipv6 multicast source-timeout** followed by the new value. For example, to set the source timeout to 360 seconds on the system if no VLAN is specified, you would enter:

```
-> ipv6 multicast source-timeout 60
```

You can also modify the source timeout on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 source-timeout 60
```

Restoring the Source Timeout

To restore the source timeout to its default (i.e., 30 seconds) value on the system if no VLAN is specified, use the **ipv6 multicast source-timeout** command by entering:

```
-> ipv6 multicast source-timeout 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast source-timeout
```

To restore the source timeout to its default value.

You can also restore the source timeout on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 source-timeout 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 source-timeout
```

To restore the source timeout to its default value.

Enabling and Disabling the MLD Querying

By default MLD querying is disabled. The following subsections describe how to enable and disable MLD querying by using the **ipv6 multicast querying** command.

Enabling the MLD Querying

You can enable the MLD querying by entering **ipv6 multicast querying** followed by the **enable** keyword. For example, to enable the MLD querying you would enter:

```
-> ipv6 multicast querying enable
```

You can also enable the MLD querying on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 querying enable
```

Disabling the MLD Querying

You can disable the MLD querying by entering **ipv6 multicast querying** followed by the **disable** keyword. For example, to disable the MLD querying you would enter:

```
-> ipv6 multicast querying disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast querying
```

To restore the MLD querying to its default setting (i.e., disabled).

You can also disable the MLD querying on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 querying disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 querying
```

To restore the MLD querying to its default setting (i.e., disabled).

You can remove an MLD querying entry on the specified VLAN and return to its default behavior by entering:

```
-> no ipv6 multicast vlan 2 querying
```

Modifying the MLD Robustness Variable

The default value of the robustness variable (i.e., the variable that allows fine-tuning on the network, where the expected packet loss is greater) is 2. The following subsections describe how to set the value of the MLD robustness variable and restore it by using the **ipv6 multicast robustness** command.

Configuring the MLD Robustness Variable

You can modify the MLD robustness variable from 1 to 7 on the system if no vlan is specified, by entering **ipv6 multicast robustness**, followed by the new value. For example, to set the value of robustness to 3 you would enter:

```
-> ipv6 multicast robustness 3
```

Note. If the links are known to be lossy, then robustness can be set to a higher value (7).

You can also modify the MLD robustness variable from 1 to 7 on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 robustness 3
```

Restoring the MLD Robustness Variable

You can restore the MLD robustness variable to its default (i.e., 2) value on the system if no vlan is specified by entering **ipv6 multicast robustness** followed by the value 0, as shown below:

```
-> ipv6 multicast robustness 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast robustness
```

To restore the MLD robustness to its default value.

You can also modify the MLD robustness variable from 1 to 7 on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 robustness 0
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 robustness
```

To restore the MLD robustness to its default value.

Enabling and Disabling the MLD Spoofing

By default, MLD spoofing (i.e., replacing a client's MAC and IPv6 address with the system's MAC and IPv6 address, when proxying aggregated MLD group membership information) is disabled on the switch. The following subsections describe how to enable and disable spoofing by using the **ipv6 multicast spoofing** command.

Enabling the MLD Spoofing

To enable MLD spoofing on the system if no VLAN is specified, you use the **ipv6 multicast spoofing** command as shown below:

```
-> ipv6 multicast spoofing enable
```

You can also enable MLD spoofing on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 spoofing enable
```

Disabling the MLD Spoofing

To disable MLD spoofing on the system if no VLAN is specified, you use the **ipv6 multicast spoofing** command as shown below:

```
-> ipv6 multicast spoofing disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast spoofing
```

To restore the MLD spoofing to its default setting (i.e., disabled).

You can also disable MLD spoofing on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 spoofing disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 spoofing
```

To restore the MLD spoofing to its default setting (i.e., disabled).

You can remove an MLD spoofing entry on the specified VLAN and return to its default behavior by entering:

```
-> no ipv6 multicast vlan 2 spoofing
```

Enabling and Disabling the MLD Zapping

By default MLD (i.e., processing membership and source filter removals immediately without waiting for the protocol's specified time period – this mode facilitates IP TV applications looking for quick changes

between IP multicast groups.) is disabled on a switch. The following subsections describe how to enable and disable zapping by using the **ipv6 multicast zapping** command.

Enabling the MLD Zapping

To enable MLD zapping on the system if no VLAN is specified, use the **ipv6 multicast zapping** command as shown below:

```
-> ipv6 multicast zapping enable
```

You can also enable MLD zapping on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 zapping enable
```

Disabling the MLD Zapping

To disable MLD zapping on the system if no VLAN is specified, use the **ipv6 multicast zapping** command as shown below:

```
-> ipv6 multicast zapping disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast zapping
```

To restore the MLD zapping to its default setting (i.e., disabled).

You can also disable MLD zapping on the specified VLAN by entering:

```
-> ipv6 multicast vlan 2 zapping disable
```

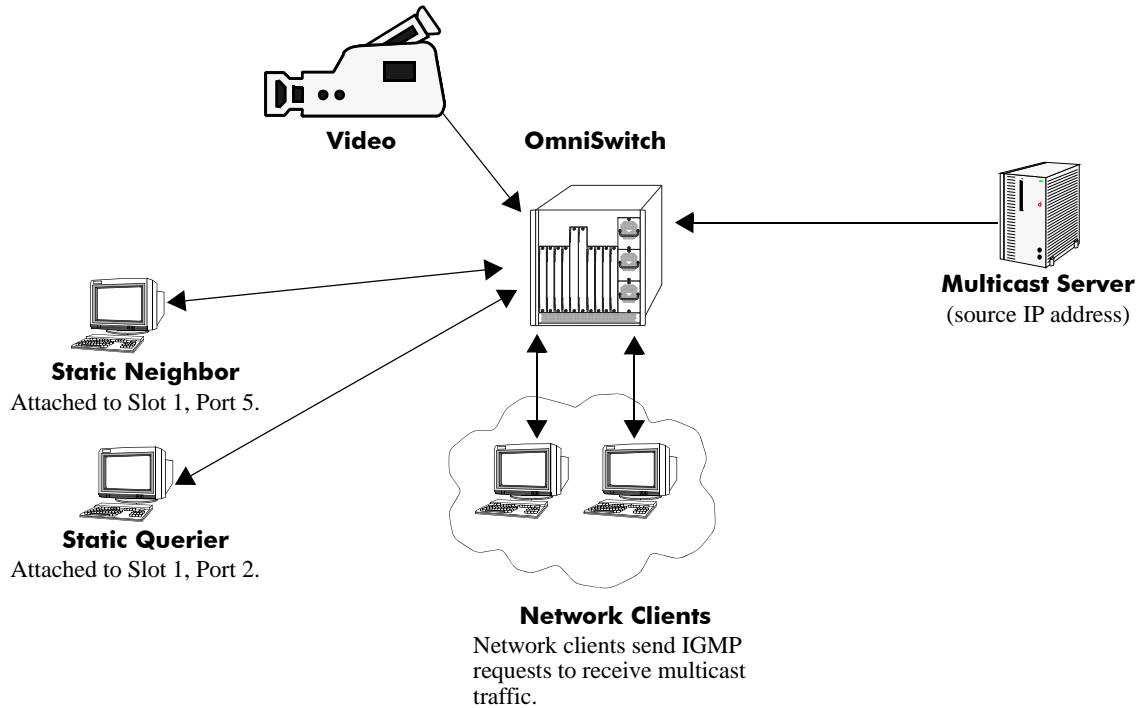
Or, as an alternative, enter:

```
-> ipv6 multicast vlan 2 zapping
```

To restore the MLD zapping to its default setting (i.e., disabled).

IPMS Application Example

The figure below shows a sample network with the switch sending multicast video. A client attached to Port 5 needs to be configured as a static IGMP neighbor and another client attached to Port 2 needs to be configured as a static IGMP querier.



Example of IPMS Network

The network administrator has determined that the network is too lossy and therefore the robustness variable needs to be set to a higher (i.e., 7) value.

Follow the steps below to configure this network:

Note. All the steps following Step 1 (which must be executed first) may be entered in any order.

1 Enable IP Multicast Switching switch-wide, by entering:

```
-> ip multicast status enable
```

2 Configure the client attached to Port 5 as a static neighbor belonging to VLAN 5 by entering:

```
-> ip multicast static-neighbor vlan 5 port 1/5
```

3 Configure the client attached to Port 2 as a static querier belonging to VLAN 5 by entering:

```
-> ip multicast static-querier vlan 5 port 1/2
```

4 Modify the robustness variable from its default value of 2 to 7 by entering:

```
-> ip multicast robustness 7
```

An example of what these commands look like entered sequentially on the command line:

```
-> ip multicast status enable
-> ip multicast static-neighbor vlan 5 port 1/5
-> ip multicast static-querier vlan 5 port 1/2
-> ip multicast robustness 7
```

As an option, you can use the **show ip multicast**, **show ip multicast neighbor**, and **show ip multicast querier** commands to confirm your settings as shown below:

```
-> show ip multicast
```

```
Status: = Enabled
Querying: = Disabled
Proxying: = Disabled
Spoofing: = Disabled
Zapping: = Disabled
Querier Forwarding: = Disabled
Version: = 1
Robustness: = 2
Query Interval (seconds): = 125
Query Response Interval (milliseconds): = 10000
Last Member Query Interval(milliseconds): = 1000
Unsolicited Report Interval (seconds) = 1,
Router Timeout (seconds): = 90
Source Timeout (seconds): = 30
```

```
-> show ip multicast neighbor
```

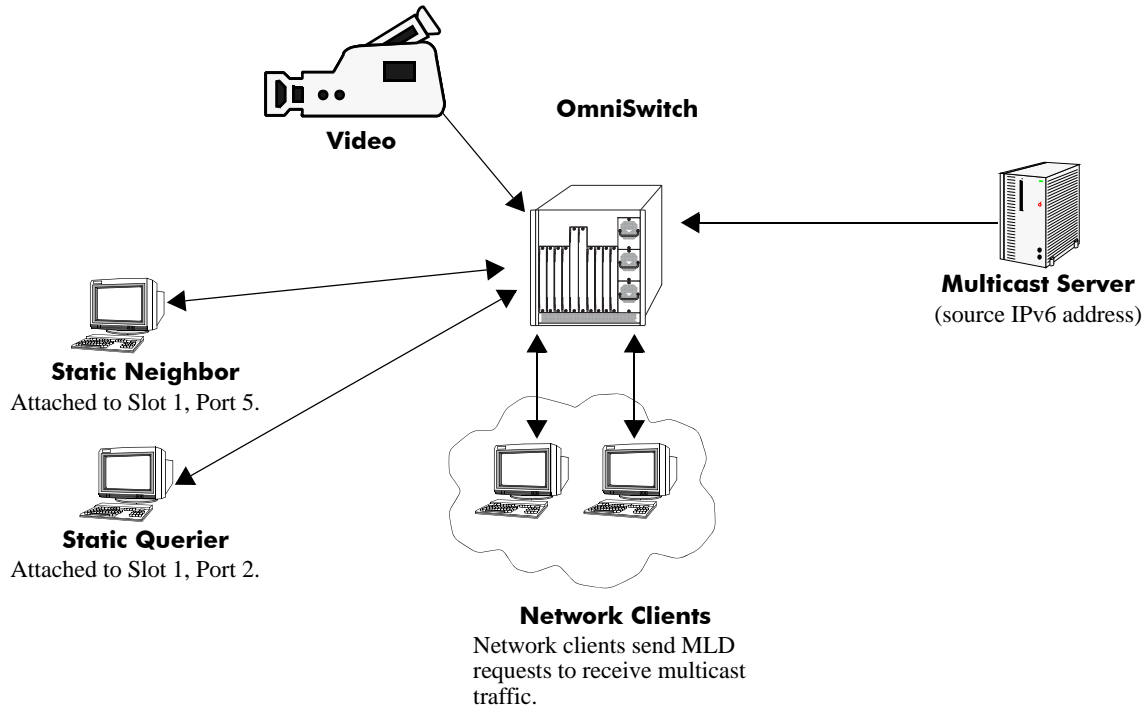
```
Total 1 Neighbors
Host Address      VLAN  Port  Static  Count  Life
-----+-----+-----+-----+-----
1.0.0.2           5    1/5   no      1      86
```

```
-> show ip multicast querier
```

```
Total 1 Queriers
Host Address      VLAN  Port  Static  Count  Life
-----+-----+-----+-----+-----
1.0.0.3           5    1/2   no      1     250
```

IPMSv6 Application Example

The figure below shows a sample network with the switch sending multicast video. A client attached to Port 5 needs to be configured as a static MLD neighbor and another client attached to Port 2 needs to be configured as a static MLD querier.



Example of IMPS Network

The network administrator has determined that the network is too lossy and therefore the robustness variable needs to be set to a higher (i.e., 7) value.

Follow the steps below to configure this network:

Note. All the steps following Step 1 (which must be executed first) may be entered in any order.

1 Enable IP Multicast Switching switch-wide, by entering:

```
-> ipv6 multicast status enable
```

2 Configure the client attached to Port 5 as a static MLD neighbor belonging to VLAN 5 by entering:

```
-> ipv6 multicast static-neighbor vlan 5 port 1/5
```

3 Configure the client attached to Port 2 as a static MLD querier belonging to VLAN 5 by entering:

```
-> ipv6 multicast static-querier vlan 5 port 1/2
```

4 Modify the robustness variable from its default value of 2 to 7 by entering:

```
-> ipv6 multicast robustness 7
```


An example of what these commands look like entered sequentially on the command line:

```
-> ipv6 multicast status enable
-> ipv6 multicast static-neighbor vlan 5 port 1/5
-> ipv6 multicast static-querier vlan 5 port 1/2
-> ipv6 multicast robustness 7
```

As an option, you can use the **show ipv6 multicast**, **show ipv6 multicast neighbor**, and **show ipv6 multicast querier** commands to confirm your settings as shown below:

```
-> show ipv6 multicast
```

```
Status: = Enabled
Querying: = Disabled
Proxying: = Disabled
Spoofing: = Disabled
Zapping: = Disabled
Querier Forwarding: = Disabled
Version: = 1
Robustness: = 2
Query Interval (seconds): = 125
Query Response Interval (milliseconds): = 10000
Last Member Query Interval(milliseconds): = 1000
Unsolicited Report Interval (seconds) = 1,
Router Timeout (seconds): = 90
Source Timeout (seconds): = 30
```

```
-> show ipv6 multicast neighbor
```

```
Total 1 Neighbors
Host Address          VLAN  Port  Static  Count  Life
-----+-----+-----+-----+-----
fe80::2a0:ccff:fed3:2853  5    1/5   no      1      6
```

```
-> show ipv6 multicast querier
```

```
Total 1 Queriers
Host Address          VLAN  Port  Static  Count  Life
-----+-----+-----+-----+-----
fe80::2a0:ccff:fed3:2854  5    1/2   no      1      6
```

Displaying IPMS Configurations and Statistics

Alcatel-Lucent's IP Multicast Switching (IPMS) **show** commands provide tools to monitor IPMS traffic and settings and to troubleshoot problems. These commands are described below:

show ip multicast	Displays the general IP Multicast switching configuration parameters on a switch.
show ip multicast group	Displays all detected multicast groups that have members. If you do not specify an IP address then all multicast groups on the switch will be displayed.
show ip multicast neighbor	Displays all neighboring multicast routers.
show ip multicast querier	Displays all multicast queriers.
show ip multicast forward	Displays the IPMS multicast forwarding table. If you do not specify a multicast group IP address, then the forwarding table for all multicast groups will be displayed.
show ip multicast source	Displays the IPMS multicast source table. If you do not specify a multicast group IP address, then the source table for all multicast groups will be displayed.
show ip multicast tunnel	Displays the IP multicast switch tunneling table entries matching the specified IP multicast group address, or all the entries if no IP multicast address is specified.

If you are interested in a quick look at IPMS groups on your switch you could use the **show ip multicast group** command. For example:

```
-> show ip multicast group
```

```
Total 3 Groups
Group Address      Source Address  VLAN  Port  Mode      Static  Count  Life
-----+-----+-----+-----+-----+-----+-----+-----
231.0.0.3         1.0.0.5        1     2/1  exclude  no      1     257
234.0.0.4         0.0.0.0        1     2/1  exclude  no      1     218
229.0.0.1         0.0.0.0        1     2/13 exclude  yes     0      0
```

Note. See the “IP Multicast Switching Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation on IPMS **show** commands.

Displaying IPMSv6 Configurations and Statistics

Alcatel-Lucent's IPv6 Multicast Switching (IPMSv6) **show** commands provide tools to monitor IPMSv6 traffic and settings and to troubleshoot problems. These commands are described below:

- show ipv6 multicast** Displays the general IPv6 Multicast switching configuration parameters on a switch.
- show ipv6 multicast group** Displays all detected multicast groups that have members. If you do not specify an IPv6 address, then all multicast groups on the switch will be displayed.
- show ipv6 multicast neighbor** Displays all neighboring IPv6 multicast routers.
- show ipv6 multicast querier** Displays all IPv6 multicast queriers.
- show ipv6 multicast forward** Displays the IPMSv6 multicast forwarding table. If you do not specify a multicast group IPv6 address, then the forwarding table for all multicast groups will be displayed.
- show ipv6 multicast source** Displays the IPMSv6 multicast source table. If you do not specify a multicast group IPv6 address, then the source table for all multicast groups will be displayed.

If you are interested in a quick look at IPMSv6 groups on your switch you could use the **show ipv6 multicast group** command. For example:

```
-> show ipv6 multicast group
```

```
Total 3 Groups
Group Address      Source Address    VLAN  Port  Mode      Static  Count  Life
-----+-----+-----+-----+-----+-----+-----+-----
ff05::5           ::               1     2/1  exclude  no      1     145
ff05::6           3333::1         1     2/1  exclude  no      1     242
ff05::9           ::               1     2/13 exclude  yes     0      0
```

Note. See the “IPv6 Multicast Switching Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation on IPMS **show** commands.

37 Configuring IP Multicast VLAN

Multicasting is a one-to-many transmission mode. It is similar to broadcasting, except that multicasting means sending to specific groups, whereas broadcasting implies sending to all. When sending voluminous data, multicast saves considerable bandwidth as the bulk of the data is transmitted only once from its source through major backbones and are distributed out at switching points closer to end users.

IP Multicast VLAN (IPMV) is an innovative feature for service providers delivering residential voice and video services. It involves the creation of separate dedicated VLANs built specifically for multicast traffic distribution. These distribution VLANs connect to the nearest multicast router and support multicast traffic only.

In This Chapter

This chapter describes the basic components of IP Multicast VLAN and shows how to configure them through the Command Line Interface (CLI). CLI commands are used in configuration examples; for more details about command syntax, see the *OmniSwitch 6250 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Creating and Deleting IPMVLAN on [page 37-9](#).
- Assigning and Deleting IPv4/IPv6 Addresses on [page 37-10](#).
- Assigning and Deleting a C-Tag on [page 37-10](#).
- Creating and Deleting a Sender Port on [page 37-11](#).
- Creating and Deleting a Receiver Port on [page 37-11](#).
- Associating an IPMVLAN with a Customer VLAN on [page 37-12](#).

Note. You can also configure and monitor IPMV through WebView, Alcatel-Lucent's embedded web-based device management application. WebView is an interactive and easy-to-use GUI that can be launched from OmniVista or a web browser. Please refer to WebView's online documentation for more information on configuring and monitoring IPMV through WebView.

IP Multicast VLAN Specifications

The following table lists IPMVLAN specifications.

IEEE Standards Supported	802.1ad/D6.0 Standard for Local and Metropolitan Area Networks - Virtual Bridged Local Area Networks - Amendment 4: Provider Bridges
Platforms Supported	OmniSwitch 6250 Series
Maximum Number of IP Multicast VLAN IDs	256 (The valid range is 2 through 4094)
VLAN Stacking Functionality Modes	VLAN Stacking mode Enterprise mode

IP Multicast VLAN Defaults

The following table lists IPMVLAN default values.

Parameter Description	Command	Default Value/Comments
Administrative Status	<code>vlan ipmvlan</code>	Enabled

IP Multicast VLAN Overview

The IP Multicast VLAN (IPMV) feature helps service providers to create separate dedicated VLANs to distribute multicast traffic. Service providers have to separate users using these VLANs. This should be done along with the distribution of broadcast media through IP Multicast across these VLANs without a router in the distribution L2 switch. To achieve this, the distribution L2 switch needs to perform IGMP snooping (i.e., allow the switch to "listen in" on the IGMP conversation between hosts and routers) as well as distribute multicast traffic from one multicast distribution VLAN to many customer ports.

A distribution multicast VLAN that switches into customer ports is invisible to the customer to avoid packet duplication across the trunk. Furthermore, some service providers use QinQ on the provider ports to tag the multicast distribution VLAN with a distinct outer VLAN tag. The customer ports can either be tagged or untagged. However, the multicast traffic always needs to be tagged. This process requires one or more separate multicast distribution VLANs. These distribution VLANs connect to the nearest multicast router and are used for multicast traffic only.

The multicast traffic will only flow from the distribution VLAN to the customer VLAN. Customer-generated multicast traffic will flow only through the customer VLANs so that the multicast router can control the distribution of such traffic.

The IPMV feature works in both the Enterprise and the VLAN Stacking environment. The ports are classified as VLAN Stacking ports and Legacy ports (fixed ports/tagged ports). To ascertain that data flow is limited to either the VLAN Stacking domain or the Enterprise domain, VLAN Stacking ports must be members of VLAN Stacking VLANs only, while the normal Legacy ports must be members of VLANs configured in the Enterprise mode only.

It is not possible to change an IPMVLAN from one mode to another. An IPMVLAN configured in a specific mode must first be deleted, then re-created in the other mode.

VLAN Stacking Mode

IP Multicast VLANs in the VLAN Stacking mode contain VLAN Stacking ports as their member ports. In an IPMVLAN, the VLAN Stacking network port (NNI) corresponds to the sender port, which also receives multicast data for the configured multicast group. Only one sender port can be assigned to an IPMVLAN. The VLAN Stacking user port (UNI) corresponds to the receiver port of the IPMVLAN. An IPMVLAN can include multiple receiver ports as its members.

IPMVLAN Lookup Mode

In the VLAN Stacking double-tagged mode, single-tagged IGMP reports are double-tagged and sent to the CPU of the Ethernet switch.

The IP Multicast Switching (IPMS) module can use any one of the following methods to bind IPMVLANs to a single receiver port:

- IP address, or
- CVLAN-tag, received as part of the IGMP report

Note. It is recommended to use any one of the methods on the receiver port and not both.

Note. CVLAN-tag translation rule applies only in the VLAN Stacking mode.

You can use the **vlan ipmvlan ctag** command to define the translation rule for replacing the outer s-tag with an IPMVLAN ID, the inner being the customer tag (c-tag).

Note. No checks will be performed on c-tags as they are simple translation rules. VLAN addition or deletion rules do not affect them.

The following limitations should be noted in the c-tag translation mode:

- The translation rule applies only to double-tagged frames.
- IP address translation rule applies to untagged IGMP reports received from customer.
- The translation rule applies only to the VLAN Stacking IPMVLANs.

Enterprise Mode

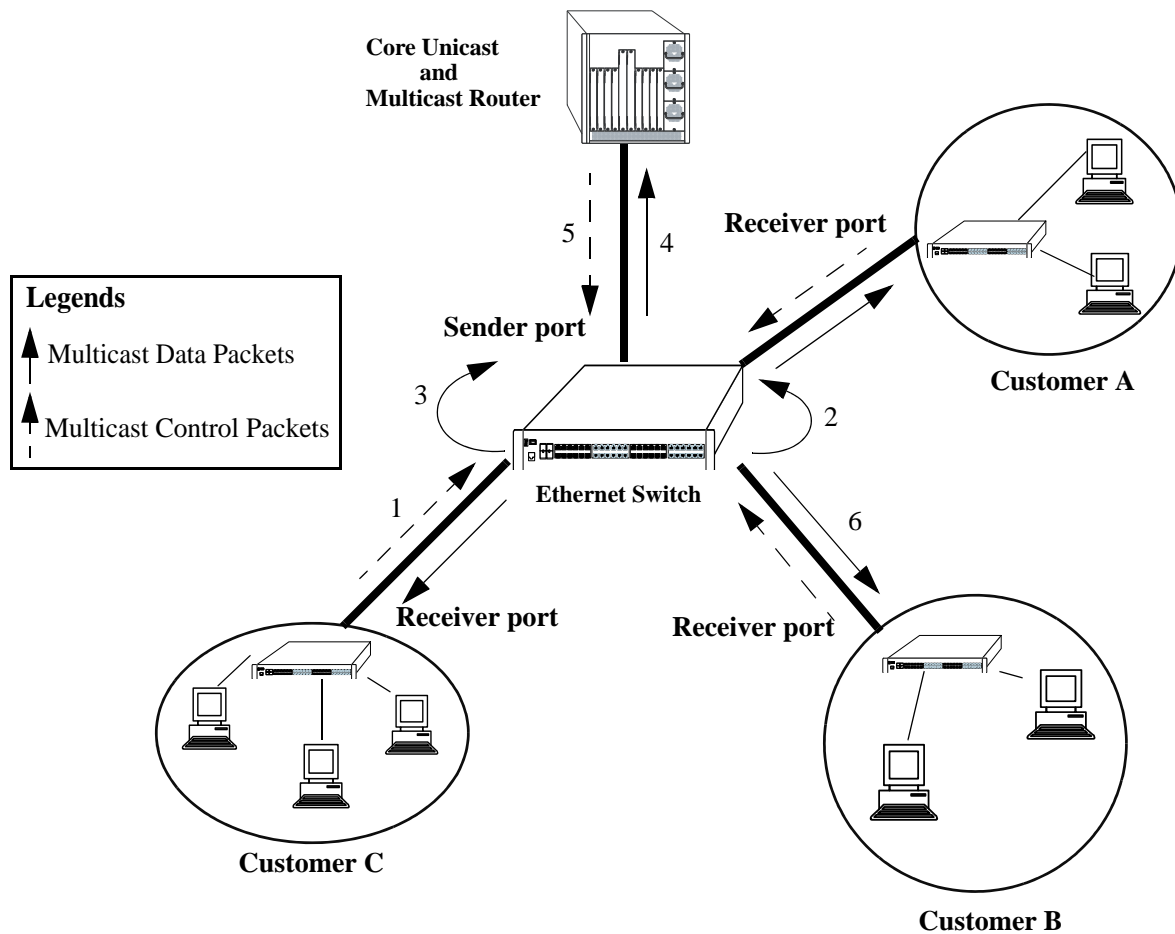
IP Multicast VLANs in the Enterprise mode contain normal user ports (fixed/tagged) as their member ports.

IPMV Packet Flows

This section describes the tagged and untagged packet flows in both the Enterprise and VLAN Stacking modes. In addition, it also describes the packet flow from the ingress point to the egress point.

VLAN Stacking Mode

The following illustration shows customers A, B, and C formed as a multicast group G1. Three types of control packets ingress on the receiver port.



Packet Flow in the VLAN Stacking Mode

The paths taken by the packets are described in the following subsections:

Untagged Control Packets Ingressing on the Receiver Port

The following steps describe the path taken by untagged control packets ingressing on the receiver port:

- 1** Untagged IPMS join reports for the multicast group G1 are sent to the receiver port.
- 2** The IPMS reports sent to the CPU of the Ethernet switch are single-tagged with the default SVLAN tag (s-tag).
- 3** IPMS overwrites the SVLAN tag with the IPMV tag after IPMV table lookup.
- 4** A single IPMS report, single-tagged with IPMV, is sent to the multicast server for group G1.
- 5** The single multicast data packet, single-tagged with IPMV, is generated by the multicast server for group G1.
- 6** The generated multicast data packets are flooded on the receiver port. These data packets are untagged.

C-Tag Translation Rule in the VLAN Stacking Mode

The following steps describe how the c-tag translation rule works in the VLAN Stacking mode:

- 1** The IPMS join reports for multicast group G1, which are single-tagged with the CVLAN tag (c-tag) are sent to the receiver port.
- 2** SVLAN tags are attached before the CVLAN tags in all the IPMS reports going to the CPU of the Ethernet switch.
- 3** IPMS overwrites the SVLAN tags with the IPMV tags after IPMV table lookup for the inner c-tag.
- 4** A single IPMS double-tagged report with an IPMV outer tag and a CVLAN inner tag is sent to the multicast server for group G1.
- 5** The single multicast double-tagged data packets with an IPMV outer tag and a CVLAN inner tag are generated by the multicast server for group G1.
- 6** The VLAN Stacking egress logic removes the IPMV outer tag. The generated multicast data packets flooded on the receiver port are single-tagged with CVLAN.

Single-Tagged Control Packets (with CVLAN) Ingressing on the Receiver Port in the VLAN Stacking Double-Tag Mode

The following steps describe the path taken by single-tagged control packets ingressing on the receiver port in the VLAN Stacking double-tag mode:

- 1** The IPMS join reports for multicast group G1, single-tagged with the CVLAN tag (c-tag), are sent to the receiver.
- 2** SVLAN tags are attached after the CVLAN tags in all the IPMS reports going to the CPU of the Ethernet switch.
- 3** IPMS overwrites the SVLAN tags with the IPMV tags after IPMV table lookup for the inner c-tag.
- 4** A single IPMS double-tagged report with an IPMV outer tag and a CVLAN inner tag is sent to the multicast server for group G1.

- 5** The single multicast double-tagged data packets with an IPMV outer tag and a CVLAN inner tag are generated by the multicast server for group G1.
- 6** The VLAN Stacking egress logic removes the IPMV outer tag. The generated multicast data packets flooded on the receiver port are single-tagged with CVLAN.

Note. All the IPMS control traffic specified for a single multicast service should be tagged with the same CVLAN.

Single-Tagged Control Packets (with CVLAN) Ingressing on the Receiver Port in the VLAN Stacking Translation Mode

The following steps describe the path taken by single-tagged control packets ingressing on the receiver port in the VLAN Stacking translation mode:

- 1** The IPMS join reports for multicast group G1, which are single-tagged with the CVLAN tag (c-tag) are sent to the receiver port.
- 2** CVLAN tags are replaced by the SVLAN tags in all the IPMS reports going to the CPU of the Ethernet switch.
- 3** IPMS overwrites the SVLAN tags with the IPMV tags after IPMV table lookup.
- 4** A single IPMV-tagged IPMS report is sent to the multicast server for Group G1.
- 5** The single multicast packets single-tagged with IPMV are generated by the multicast server for group G1.
- 6** The VLAN Stacking egress logic replaces the IPMV tag with the CVLAN tag. The multicast data packets flooded on the receiver port are single-tagged with CVLAN.

Note. All the IPMS control traffic specified for a single multicast service should be tagged with the same CVLAN.

Enterprise Mode

In the Enterprise mode, two types of control packets ingress on the receiver ports. The paths taken by the packets (as shown in the diagram on [page 37-5](#)) are described in the following subsections.

Untagged Control Packets Ingressing on the Receiver Port

The following steps describe the path taken by untagged control packets ingressing on the receiver port:

- 1 Untagged IPMS join reports for the multicast group G1 are sent to the receiver port.
- 2 The IPMS reports sent to the CPU of the Ethernet switch are single-tagged with the default VLAN.
- 3 IPMS overwrites the tag with the IPMV tag after IPMV table lookup.
- 4 A single IPMS report, single-tagged with IPMV, is sent to the multicast server for group G1.
- 5 The single multicast data packet, single-tagged with IPMV, is generated by the multicast server for group G1.
- 6 The generated multicast data packets are flooded on the receiver port. These data packets are untagged.

Tagged Control Packets Ingressing on the Receiver Port

The following steps describe the path taken by tagged control packets ingressing on the receiver port:

- 1 The single-tagged IPMS join reports for the multicast group G1 are sent to the receiver port.
- 2 The IPMS reports are sent to the CPU of the Ethernet switch.
- 3 IPMS overwrites the tag with the IPMV tag after IPMV table lookup.
- 4 A single IPMS report, single-tagged with IPMV, is sent to the multicast server for group G1.
- 5 The single multicast data packet, single-tagged with IPMV, is generated by the multicast server for group G1.
- 6 The generated multicast data packets are flooded on the receiver port. These data packets are untagged.

Configuring IPMVLAN

This section describes how to use Command Line Interface (CLI) commands to complete the following configuration tasks:

- Creating and deleting IPMVLAN (see [“Creating and Deleting IPMVLAN” on page 37-9](#)).
- Assigning IPv4/IPv6 address to an existing IPMVLAN and removing it (see [“Assigning and Deleting IPv4/IPv6 Address” on page 37-10](#)).
- Assigning and removing the c-tag in an IPMVLAN (see [“Assigning and Deleting a Customer VLAN Tag” on page 37-10](#)).
- Creating and deleting a sender port in an IPMVLAN (see [“Creating and Deleting a Sender Port” on page 37-11](#)).
- Creating and deleting a receiver port in an IPMVLAN (see [“Creating and Deleting a Receiver Port” on page 37-11](#)).
- Configuring a VLAN translation of a CVLAN to an IPMVLAN (see [“Associating an IPMVLAN with a Customer VLAN” on page 37-12](#)).

In addition, a tutorial is provided in [“IPMVLAN Application Example” on page 37-13](#) that shows you how to use CLI commands to configure a sample network.

Note. See the “IP Multicast VLAN Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide* for complete documentation of IPMVLAN CLI commands.

Creating and Deleting IPMVLAN

The following subsections describe how to create and delete an IPMVLAN with the [vlan ipmvlan](#) command.

Note. The Enterprise mode is the default mode of an IP Multicast VLAN.

Creating IPMVLAN

To create an IPMVLAN, use the [vlan ipmvlan](#) command as shown below:

```
-> vlan ipmvlan 1003 name
"multicast vlan"
```

For example, to create an IPMVLAN in the 1x1 Spanning Tree mode, enter:

```
-> vlan ipmvlan 1333 1x1 stp enable name "nvlan"
```

Deleting IPMVLAN

To remove an IPMVLAN, use the **no** form of the **vlan ipmvlan** command by entering **no vlan ipmvlan** followed by the IPMVLAN ID, as shown below:

```
-> no vlan ipmvlan 1003
```

To remove multiple IPMVLANs, specify a range of IPMVLAN IDs. For example:

```
-> no vlan ipmvlan 1010-1017
```

Assigning and Deleting IPv4/IPv6 Address

The following subsections describe how to assign an IPv4 or IPv6 address to an existing IPMVLAN as well as delete the same with the **vlan ipmvlan address** command.

Assigning an IPv4/IPv6 Address to an IPMVLAN

To assign an IPv4 or IPv6 address to an existing IPMVLAN, use the **vlan ipmvlan address** command as shown below:

```
-> vlan ipmvlan 1003 address 225.0.0.1  
-> vlan ipmvlan 1033 address ff08::3
```

Deleting an IPv4/IPv6 Address from an IPMVLAN

To delete an IPv4 or IPv6 address from an existing IP Multicast VLAN, use the **no** form of the **vlan ipmvlan address** command by entering **no vlan ipmvlan** followed by the IPMVLAN ID, the keyword **address**, and either the IPv4 or IPv6 address, as shown below:

```
-> no vlan ipmvlan 1003 address 225.0.0.1  
-> no vlan ipmvlan 1033 address ff08::3
```

Assigning and Deleting a Customer VLAN Tag

The following subsections describe how to assign and delete a customer VLAN tag (c-tag) in an IPMVLAN using the **vlan ipmvlan ctag** command.

Assigning C-Tag to an IPMVLAN

To assign c-tag to an IP Multicast VLAN, use the **vlan ipmvlan ctag** command as shown below:

```
-> vlan ipmvlan 1003 ctag 10
```

Deleting C-Tag from an IPMVLAN

To delete c-tag from an IPMVLAN, use the **no** form of the **vlan ipmvlan ctag** command by entering **no vlan ipmvlan** followed by the IPMVLAN ID, the keyword **ctag**, and the customer VLAN ID number, as shown below:

```
-> no vlan ipmvlan 1003 ctag 10
```

Creating and Deleting a Sender Port

The following subsections describe how to create and delete a sender port in an IPMVLAN with the `vlan ipmvlan sender-port` command.

Creating a Sender Port in an IPMVLAN

To create a sender port in an IPMVLAN configured in the Enterprise mode, use the `vlan ipmvlan sender-port` command as shown below:

```
-> vlan ipmvlan 1003 sender-port port 1/50
```

To create multiple sender ports in an IPMVLAN, specify a range of ports. For example:

```
-> vlan ipmvlan 1003 sender-port port 1/45-48
```

In the VLAN Stacking mode, the port that you want to configure as a sender port should be a VLAN Stacking port (network port). To create a sender port in an IPMVLAN configured in the VLAN Stacking mode, use the `vlan ipmvlan sender-port` command as shown below:

```
-> vlan svlan 1/49 network-port  
-> vlan ipmvlan 1033 sender-port port 1/49
```

Deleting a Sender Port from an IPMVLAN

To delete a sender port from an IPMVLAN in the Enterprise or VLAN Stacking mode, use the **no** form of the `vlan ipmvlan sender-port` command by entering **no vlan ipmvlan** followed by the IPMVLAN ID, the keyword **sender-port**, and the port number, as shown below:

```
-> no vlan ipmvlan 1003 sender-port port 1/50
```

The following command deletes multiple sender ports from an IPMVLAN:

```
-> no vlan ipmvlan 1003 sender-port port 1/45-48
```

Creating and Deleting a Receiver Port

The following subsections describe how to create and delete a receiver port in an IPMVLAN with the `vlan ipmvlan receiver-port` command.

Creating a Receiver Port in an IPMVLAN

To create a receiver port in an IPMVLAN configured in the Enterprise mode, use the `vlan ipmvlan receiver-port` command as shown below:

```
-> vlan ipmvlan 1003 receiver-port port 1/51
```

In the VLAN Stacking mode, the port you want to configure as a receiver port should be a VLAN Stacking user port (UNI). To create a receiver port in an IPMVLAN configured in the VLAN Stacking mode, use the `vlan ipmvlan receiver-port` command as shown below:

```
-> vlan svlan port 1/1 user-customer-port default-svlan 10  
-> vlan ipmvlan 1002 receiver-port port 1/1
```

Deleting a Receiver Port from an IPMVLAN

To delete a receiver port from an IPMVLAN in the Enterprise or VLAN Stacking mode, use the **no** form of the **vlan ipmvlan receiver-port** command by entering **no vlan ipmvlan** followed by the IPMVLAN ID, the keyword **receiver-port**, and the port number, as shown below:

```
-> no vlan ipmvlan 1003 receiver-port port 1/51
```

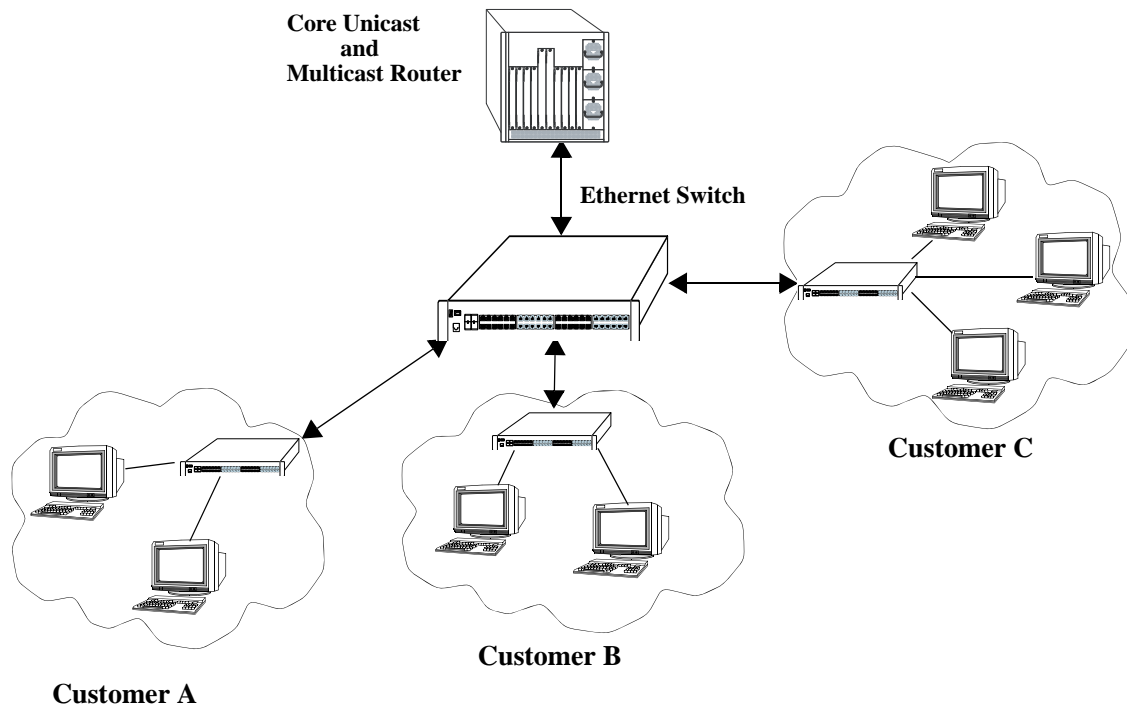
Associating an IPMVLAN with a Customer VLAN

To associate an IPMVLAN with a customer VLAN, use the **vlan svlan port translate ipmvlan** command. Note that the port you want to use to associate an IPMVLAN with a customer VLAN should be a receiver port. Also, the receiver port must be a VLAN Stacking user port (UNI). For example, the following series of commands will associate an IPMVLAN with a customer VLAN:

```
-> vlan svlan port 1/1 user-customer-port default-svlan 10  
-> vlan ipmvlan 1002 receiver-port port 1/1  
-> vlan svlan port 1/1 translate cvlan 10 ipmvlan 1002
```


IPMVLAN Application Example

The figure below shows a sample IPMVLAN network with three customers A, B, and C, respectively. The customers are connected to the Ethernet switch requesting multicast data.



Example of an IPMVLAN Network

Follow the steps below to configure this network:

Note. All the steps following step 1 (which must be executed first) may be entered in any order.

1 Create an IPMVLAN by entering:

```
-> vlan ipmvlan 1003 name "multicast vlan"
```

2 Assign IPv4/IPv6 address to the IPMVLAN by entering:

```
-> vlan ipmvlan 1003 address 225.0.0.1
```

3 Create a sender port in the Enterprise mode of IPMVLAN by entering:

```
-> vlan ipmvlan 1003 sender-port port 1/50
```

Alternatively, a sender port can also be created in the VLAN Stacking mode by entering:

```
-> vlan svlan 1/49 network-port 1/49
-> vlan ipmvlan 1033 sender-port port 1/49
```

4 Create a receiver port in the Enterprise mode of IPMVLAN by entering:

```
-> vlan ipmvlan 1003 receiver-port port 1/51-60
```

Alternatively, a receiver port can also be created in the VLAN Stacking mode by entering:

```
-> vlan svlan port 1/1 user-customer-port default-svlan 10
-> vlan ipmvlan 1002 receiver-port port 1/1
```

An example of what these commands look like when entered sequentially on the command line:

```
-> vlan ipmvlan 1003 name "multicast vlan"
-> vlan ipmvlan 1003 address 225.0.0.1
-> vlan ipmvlan 1003 sender-port port 1/50
-> vlan ipmvlan 1003 receiver-port port 1/51-60
```

As an option, you can use the [show vlan ipmvlan c-tag](#), [show vlan ipmvlan address](#), [show vlan ipmvlan port-config](#), and [show vlan ipmvlan port-binding](#) commands to confirm your settings. For example:

```
-> show vlan
```

vlan	type	admin	stree			auth	ip	mble	
			oper	lxl	flat			tag	name
1	std	on	on	on	on	off	NA	off	VLAN 1
2	ipmtv	on	on	off	off	off	NA	off	IPMVLAN 2
3	ipmtv	on	on	off	off	off	NA	off	IPMVLAN 3
4	vstk	on	on	on	on	off	NA	off	SVLAN 4

```
-> show vlan ipmvlan 10 address
```

IpAddress	ipAddressType
224.1.1.1	Ipv4
224.1.1.2	Ipv4
224.1.1.3	Ipv4
ffae::1	Ipv6
ffae::2	Ipv6
ffae::3	Ipv6

```
-> show vlan ipmvlan 10 port-config
```

port	type
1/10	sender
1/20	receiver
1/30	receiver
1/49	receiver

Verifying the IP Multicast VLAN Configuration

To display information about IPMV, use the following commands:

show vlan ipmvlan	Displays IPMVLAN information for a specific IPMVLAN, a range of IPMVLANs, or all IPMVLANs.
show vlan ipmvlan c-tag	Displays the customer VLAN IDs associated with a single IP Multicast VLAN or all the configured IP Multicast VLANs.
show vlan ipmvlan address	Displays the IPv4 and IPv6 addresses assigned to a single IP Multicast VLAN or all the configured IP Multicast VLANs.
show vlan ipmvlan port-config	Displays the sender and receiver ports for a specific IP Multicast VLAN or all the IP Multicast VLANs.
show ipmvlan port-config	Displays the sender and receiver IPMVLANs for a specific slot or port.
show vlan ipmvlan port-binding	Displays the translation bindings of an IP Multicast VLAN on a port, an aggregate of ports, or all ports.

38 Diagnosing Switch Problems

Several tools are available for diagnosing problems that may occur with the switch. These tools include:

- Port Mirroring
- Port Monitoring
- sFlow
- Remote Monitoring (RMON) probes
- Switch Health Monitoring

Port mirroring copies all incoming and outgoing traffic from a single mirrored Ethernet port to a second mirroring Ethernet port, where it can be monitored with a Remote Network Monitoring (RMON) probe or network analysis device without disrupting traffic flow on the mirrored port. The port monitoring feature allows you to examine packets to and from a specific Ethernet port. sFlow is used for measuring high speed switched network traffic. It is also used for collecting, storing, and analyzing the traffic data. Switch Health monitoring software checks previously configured threshold levels for the switch's consumable resources, and notifies the Network Monitoring Station (NMS) if those limits are violated.

In This Chapter

This chapter describes port mirroring, port monitoring, remote monitoring (RMON) probes, sFlow, and switch health features and explains how to configure the same through the Command Line Interface (CLI).

Configuration procedures described in this chapter include:

Port Mirroring

- Creating or Deleting a Port Mirroring Session—see [“Creating a Mirroring Session”](#) on page 38-18 or [“Deleting A Mirroring Session”](#) on page 38-21.
- Protection from Spanning Tree changes (Port Mirroring)—see [“Unblocking Ports \(Protection from Spanning Tree\)”](#) on page 38-19.
- Enabling or Disabling Port Mirroring Status—see [“Enabling or Disabling Mirroring Status”](#) on page 38-19 or [“Disabling a Mirroring Session \(Disabling Mirroring Status\)”](#) on page 38-19.
- Configuring Port Mirroring Direction—see [“Configuring Port Mirroring Direction”](#) on page 38-20.
- Enabling or Disabling a Port Mirroring Session—see [“Enabling or Disabling a Port Mirroring Session \(Shorthand\)”](#) on page 38-20.

Port Monitoring

- [Configuring a Port Monitoring Session](#)—see “[Configuring a Port Monitoring Session](#)” on page 38-25.
- [Enabling a Port Monitoring Session](#)—see “[Enabling a Port Monitoring Session](#)” on page 38-25.
- [Disabling a Port Monitoring Session](#)—see “[Disabling a Port Monitoring Session](#)” on page 38-25.
- [Deleting a Port Monitoring Session](#)—see “[Deleting a Port Monitoring Session](#)” on page 38-25.
- [Pausing a Port Monitoring Session](#)—see “[Pausing a Port Monitoring Session](#)” on page 38-26.
- [Configuring the persistence of a Port Monitoring Session](#)—see “[Configuring Port Monitoring Session Persistence](#)” on page 38-26.
- [Configuring a Port Monitoring data file](#)—see “[Configuring a Port Monitoring Data File](#)” on page 38-26.
- [Suppressing creation of a Port Monitoring data file](#)—see “[Suppressing Port Monitoring File Creation](#)” on page 38-27.
- [Configuring a Port Monitoring direction](#)—see “[Configuring Port Monitoring Direction](#)” on page 38-27.
- [Displaying Port Monitoring Status and Data](#)—see “[Displaying Port Monitoring Status and Data](#)” on page 38-28.

sFlow

- [Configuring a sFlow Session](#)—see “[Configuring a sFlow Session](#)” on page 38-30.
- [Configuring a Fixed Primary Address](#)—see “[Configuring a Fixed Primary Address](#)” on page 38-31.
- [Displaying a sFlow Receiver](#)—see “[Displaying a sFlow Receiver](#)” on page 38-31.
- [Displaying a sFlow Sampler](#)—see “[Displaying a sFlow Sampler](#)” on page 38-32.
- [Displaying a sFlow Poller](#)—see “[Displaying a sFlow Poller](#)” on page 38-32.
- [Displaying a sFlow Agent](#)—see “[Displaying a sFlow Agent](#)” on page 38-33.
- [Deleting a sFlow Session](#)—see “[Deleting a sFlow Session](#)” on page 38-33.

RMON

- [Enabling or Disabling RMON Probes](#)—see “[Enabling or Disabling RMON Probes](#)” on page 38-36.

Switch Health Monitoring

- [Configuring Resource Threshold Limits \(Switch Health\)](#)—see “[Configuring Resource and Temperature Thresholds](#)” on page 38-43.
- [Configuring Sampling Intervals](#)—see “[Configuring Sampling Intervals](#)” on page 38-45.
- [Resetting Health Statistics](#)—see “[Resetting Health Statistics for the Switch](#)” on page 38-47.

For information about additional Diagnostics features such as Switch Logging and System Debugging/Memory Management commands, see [Chapter 39, “Using Switch Logging.”](#)

Port Mirroring Overview

The following sections detail the specifications, defaults, and quick set up steps for the port mirroring feature. Detailed procedures are found in [“Port Mirroring” on page 38-14](#).

Port Mirroring Specifications

Platforms Supported	OmniSwitch 6250 Series
Ports Supported	Ethernet (10 Mbps)/Fast Ethernet (100 Mbps)/Gigabit Ethernet (1 Gb/1000 Mbps)/10 Gigabit Ethernet (10 Gb/10000 Mbps).
Mirroring Sessions Supported	Two sessions supported per standalone switch and stack.
N-to-1 Mirroring Supported	1 to 24
Range of Unblocked VLAN IDs	1 to 4094.

Port Mirroring Defaults

The following table shows port mirroring default values.

Global Port Mirroring Defaults

Parameter Description	CLI Command	Default Value/Comments
Mirroring Session Creation	port mirroring source destination	No Mirroring Sessions Configured
Protection from Spanning Tree (Spanning Tree Disable)	port mirroring source destination	Spanning Tree Enabled
Mirroring Status Configuration	port mirroring source destination	Enabled
Mirroring Session Configuration	port mirroring	Enabled
Mirroring Session Deletion	port mirroring	No Mirroring Sessions Configured

Quick Steps for Configuring Port Mirroring

- 1 Create a port mirroring session. Be sure to specify the port mirroring session ID, source (*mirrored*) and destination (*mirroring*) slot/port, and unblocked VLAN ID (*optional*—protects the mirroring session from changes in Spanning Tree if the mirroring port will monitor mirrored traffic on an RMON probe belonging to a different VLAN). For example:

```
-> port mirroring 6 source 2/3-9 destination 2/10 unblocked 7
```

Note. *Optional.* To verify the port mirroring configuration, enter **show port mirroring status** followed by the port mirroring session ID number. The display is similar to the one shown below:

```
-> show port mirroring status 6
```

Session	Mirror Destination	Mirror Direction	Unblocked Vlan	Config Status	Oper Status
6.	2/10	-	NONE	Enable	On
Mirror Source					
6.	2/3	bidirectional	-	Enable	On
6.	2/4	bidirectional	-	Enable	On
6.	2/5	bidirectional	-	Enable	On
6.	2/6	bidirectional	-	Enable	On
6.	2/7	bidirectional	-	Enable	On
6.	2/8	bidirectional	-	Enable	On
6.	2/9	bidirectional	-	Enable	On

For more information about this command, see [“Displaying Port Mirroring Status” on page 38-21](#) or the [“Port Mirroring and Monitoring Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*](#).

Port Monitoring Overview

The following sections detail the specifications, defaults, and quick set up steps for the port mirroring feature. Detailed procedures are found in “[Port Monitoring](#)” on page 38-24.

Port Monitoring Specifications

Platforms Supported	OmniSwitch 6250 Series
Ports Supported	Ethernet (10 Mbps)/Fast Ethernet (100 Mbps)/Gigabit Ethernet (1 Gb/1000 Mbps)/10 Gigabit Ethernet (10 Gb/10000 Mbps).
Monitoring Sessions Supported	One per switch and/or stack.
File Type Supported	ENC file format (Network General Sniffer Network Analyzer Format)

Port Monitoring Defaults

The following table shows port mirroring default values.

Global Port Monitoring Defaults

Parameter Description	CLI Command	Default Value/Comments
Monitoring Session Creation	port monitoring source	No Monitoring Sessions Configured
Monitoring Status	port monitoring source	Disabled
Monitoring Session Configuration	port monitoring source	Disabled
Port Monitoring Direction	port monitoring source	Bidirectional
Data File Creation	port monitoring source	Enabled
Data File Size	port monitoring source	16384 Bytes
File Overwriting	port monitoring source	Enabled
Time before session is deleted	port monitoring source	0 seconds

Quick Steps for Configuring Port Monitoring

- 1 To create a port monitoring session, use the **port monitoring source** command by entering **port monitoring**, followed by the port monitoring session ID, **source**, and the slot and port number of the port to be monitored. For example:

```
-> port monitoring 6 source 2/3
```

- 2 Enable the port monitoring session by entering **port monitoring**, followed by the port monitoring session ID, **source**, the slot and port number of the port to be monitored, and **enable**. For example:

```
-> port monitoring 6 source 2/3 enable
```

- 3 *Optional.* Configure optional parameters. For example, to create a file called “monitor1” for port monitoring session 6 on port 2/3, enter:

```
-> port monitoring 6 source 2/3 file monitor1
```

Note. *Optional.* To verify the port monitoring configuration, enter **show port mirroring status**, followed by the port monitoring session ID number. The display is similar to the one shown below:

```
-> show port monitoring status
```

Session slot/port	Monitor Direction	Monitor Status	Overwrite Status	Operating	Admin
6.	2/ 3	Bidirectional	ON	ON	ON

For more information about this command, see “[Port Monitoring](#)” on page 38-24 or the “Port Mirroring and Monitoring Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

sFlow Overview

The following sections detail the specifications, defaults, and quick set up steps for the sFlow feature. Detailed procedures are found in “sFlow” on page 38-29.

sFlow Specifications

RFCs Supported	3176 - sFlow Management Information Base
Platforms Supported	OmniSwitch 6250 Series
Sampling	Sampling rate of one (1) counts all packets and 0 (zero) disables sampling.
Agent IP Address	As it need to send a fixed IP address in the data-gram, loopback0 IP address is used.

sFlow Defaults

The following table shows sFlow default values:

sFlow Defaults

Parameter Description	CLI Command	Default Value/Comments
Receiver Name	sflow receiver	Empty
Timeout Value	sflow receiver	0 seconds
IP Address	sflow receiver	32 bit address (IPv4)
Data File Size	sflow receiver	1400 Bytes
Version Number	sflow receiver	5
Destination Port	sflow receiver	6343
Receiver Index	sflow sampler	0
Packet Sampling Rate	sflow sampler	0
Sampled Packet Size	sflow sampler	128 Bytes
Receiver Index	sflow poller	0
Interval Value	sflow poller	0 seconds

Quick Steps for Configuring sFlow

Follow the steps below to create a sFlow receiver session.

- 1 To create a sFlow receiver session, use the **sflow receiver** command by entering **sflow receiver**, followed by the receiver index, name, and the address to be monitored. For example:

```
-> sflow receiver 1 name Golden address 198.206.181.3
```

- 2 *Optional.* Configure optional parameters. For example, to specify the timeout value “65535” for sFlow receiver session on address 198.206.181.3, enter:

```
-> sflow receiver 1 name Golden address 198.206.181.3 timeout 65535
```

Note. *Optional.* To verify the sFlow receiver configuration, enter **show sflow receiver**, followed by the sFlow receiver index. The display is similar to the one shown below:

```
-> show sflow receiver

Receiver 1
Name       = Golden
Address    = IP_V4 198.206.181.3
UDP Port   = 6343
Timeout    = 65535
Packet Size= 1400
DatagramVer= 5
```

For more information about this command, see “sFlow” on page 38-29 or the “sFlow Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Follow the steps below to create a sFlow sampler session.

- 1 To create a sFlow sampler session, use the **sflow sampler** command by entering **sflow sampler**, followed by the instance ID, port list, receiver, and the rate. For example:

```
-> sflow sampler 1 2/1-5 receiver 1 rate 2048
```

- 2 *Optional.* Configure optional parameters. For example, to specify the **sample-hdr-size** value “128” for sFlow sampler instance 1 on ports 2/1-5, enter:

```
-> sflow sampler 1 2/1-5 receiver 1 rate 2048 sample-hdr-size 128
```

Note. *Optional.* To verify the sFlow sampler configuration, enter **show sflow sampler**, followed by the sFlow sampler instance ID. The display is similar to the one shown below:

```
-> show sflow sampler 1

Instance  Interface  Receiver  Sample-rate  Sample-hdr-size
-----
1         2/ 1         1         2048         128
1         2/ 2         1         2048         128
1         2/ 3         1         2048         128
1         2/ 4         1         2048         128
1         2/ 5         1         2048         128
```

For more information about this command, see “sFlow” on page 38-29 or the “sFlow Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Follow the steps below to create a sFlow poller session.

- 1 To create a sFlow poller session, use the **sflow poller** command by entering **sflow poller**, followed by the instance ID, port list, receiver, and the interval. For example:

```
-> sflow poller 1 2/6-10 receiver 1 interval 30
```

Note. *Optional.* To verify the sFlow poller configuration, enter **show sflow poller**, followed by the sFlow poller instance ID. The display is similar to the one shown below:

```
-> show sflow poller
```

Instance	Interface	Receiver	Interval
1	2/ 6	1	30
1	2/ 7	1	30
1	2/ 8	1	30
1	2/ 9	1	30
1	2/10	1	30

For more information about this command, see “sFlow” on page 38-29 or the “sFlow Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Remote Monitoring (RMON) Overview

The following sections detail the specifications, defaults, and quick set up steps for the RMON feature. Detailed procedures are found in [“Remote Monitoring \(RMON\)” on page 38-34](#).

RMON Specifications

RFCs Supported	2819 - Remote Network Monitoring Management Information Base
Platforms Supported	OmniSwitch 6250 Series
RMON Functionality Supported	Basic RMON 4 group implementation –Ethernet Statistics group –History (Control and Statistics) group –Alarms group –Events group
RMON Functionality Not Supported	RMON 10 group* RMON2* –Host group –HostTopN group –Matrix group –Filter group –Packet Capture group (*An external RMON probe that includes RMON 10 group and RMON2 may be used where full RMON probe functionality is required.)
Flavor (Probe Type)	Ethernet/History/Alarm
Status	Active/Creating/Inactive
History Control Interval (seconds)	1 to 3600
History Sample Index Range	1 to 65535
Alarm Interval (seconds)	1 to 2147483647
Alarm Startup Alarm	Rising Alarm/Falling Alarm/ RisingOrFalling Alarm
Alarm Sample Type	Delta Value/Absolute
RMON Traps Supported	RisingAlarm/FallingAlarm These traps are generated whenever an Alarm entry crosses either its Rising Threshold or its Falling Threshold and generates an event configured for sending SNMP traps.

RMON Probe Defaults

The following table shows Remote Network Monitoring default values.

Global RMON Probe Defaults

Parameter Description	CLI Command	Default Value/Comments
RMON Probe Configuration	rmon probes	No RMON probes configured.

Quick Steps for Enabling/Disabling RMON Probes

1 Enable an inactive (or disable an active) RMON probe, where necessary. You can also enable or disable all probes of a particular flavor, if desired. For example:

```
-> rmon probes stats 1011 enable
```

```
-> rmon probes history disable
```

2 To verify the RMON probe configuration, enter the **show rmon probes** command, with the keyword for the type of probe. For example, to display the statistics probes, enter the following:

```
-> show rmon probes stats
```

The display is similar to the one shown below:

```

Entry  Slot/Port  Flavor  Status  Duration  System Resources
-----+-----+-----+-----+-----+-----
1011   1/11    Ethernet Active   11930:27:05  272 bytes

```

3 To view statistics for a particular RMON probe, enter the **show rmon probes** command, with the keyword for the type of probe, followed by the entry number for the desired RMON probe. For example:

```
-> show rmon probes 1011
```

The display will appear similar to the one shown below:

```

Probe's Owner: Switch Auto Probe on Slot 1, Port 11
Entry 1011
  Flavor = Ethernet, Status = Active,
  Time = 11930 hrs 26 mins,
  System Resources (bytes) = 272

```

For more information about these commands, see [“Displaying a List of RMON Probes” on page 38-37](#), [“Displaying Statistics for a Particular RMON Probe” on page 38-38](#), or the “RMON Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Switch Health Overview

The following sections detail the specifications, defaults, and quick set up steps for the switch health feature. Detailed procedures are found in [“Monitoring Switch Health” on page 38-41](#).

Switch Health Specifications

Platforms Supported	OmniSwitch 6250 Series
Health Functionality Supported	<ul style="list-style-type: none"> –Switch level CPU Utilization Statistics (percentage); –Switch/module/port level Input Utilization Statistics (percentage); –Switch/module/port level Input/Output Utilization Statistics (percentage); –Switch level Memory Utilization Statistics (percentage); –Device level (e.g., Chassis/CMM) Temperature Statistics (Celsius).
Monitored Resource Utilization Levels	<ul style="list-style-type: none"> –Most recent utilization level; –Average utilization level during last minute; –Average utilization level during last hour; –Maximum utilization level during last hour.
Resource Utilization Raw Sample Values	Saved for previous 60 seconds.
Resource Utilization Current Sample Values	Stored.
Resource Utilization Maximum Utilization Value	Calculated for previous 60 seconds and stored.
Utilization Value = 0	Indicates that none of the resources were measured for the period.
Utilization Value = 1	Indicates that a non-zero amount of the resource (less than 2%) was measured for the period.
Percentage Utilization Values	Calculated based on Resource Measured During Period/Total Capacity.
Resource Threshold Levels	Apply automatically across all levels of switch (switch/module/port).
Rising Threshold Crossing	A Resource Threshold was exceeded by its corresponding utilization value in the current cycle.
Falling Threshold Crossing	A Resource Threshold was exceeded by its corresponding utilization value in the previous cycle, but is not exceeded in the current cycle.
Threshold Crossing Traps Supported	Device, module, port-level threshold crossings.

Switch Health Defaults

The following table shows Switch Health default values.

Global Switch Health Defaults

Parameter Description	CLI Command	Default Value/Comments
Resource Threshold Limit Configuration	health threshold	80 percent
Sampling Interval Configuration	health interval	5 seconds
Switch Temperature	health threshold	50 degrees Celsius

Quick Steps for Configuring Switch Health

1 Display the health threshold limits, health sampling interval settings, and/or health statistics for the switch, depending on the parameters you wish to modify. (For best results, note the default settings for future reference.) For example:

```
-> show health threshold
```

The default settings for the command you entered will be displayed. For example:

```
Rx Threshold           = 80
TxRx Threshold        = 80
Memory Threshold      = 80
CPU Threshold         = 80
Temperature Threshold = 60
```

2 Enter the appropriate command to change the required health threshold or health sampling interval parameter settings or reset all health statistics for the switch. For example:

```
-> health threshold memory 85
```

Note. *Optional.* To verify the Switch Health configuration, enter [show health threshold](#), followed by the parameter you modified (e.g., **memory**). The display is similar to the one shown below:

```
Memory Threshold      = 85
```

For more information about this command, see [“Displaying Health Threshold Limits” on page 38-44](#) or the [“Health Monitoring Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*](#).

Port Mirroring

On chassis-based or standalone switches, you can set up port mirroring sessions between Ethernet ports within the same switch, while on stackable switches, you can set up port mirroring sessions across switches within the same stack.

Ethernet ports supporting port mirroring include 10BaseT/100BaseTX/1000BaseT (RJ-45), 1000BaseSX/LX/LH, and 10GBaseS/L (LC) connectors. When port mirroring is enabled, the active “mirrored” port transmits and receives network traffic normally, and the “mirroring” port receives a copy of all transmit and receive traffic to the active port. You can connect an RMON probe or network analysis device to the mirroring port to see an exact duplication of traffic on the mirrored port without disrupting network traffic to and from the mirrored port.

Port mirroring runs in the Chassis Management software and is supported for Ethernet (10 Mbps), Fast Ethernet (100 Mbps), Gigabit Ethernet (1000 Mbps), and 10 Gigabit Ethernet (10000 Mbps) ports. In addition, the switch supports “N-to-1” port mirroring, where up to 24 source ports can be mirrored to a single destination port.

Note the following restriction when configuring a port mirroring session:

- Two (2) port mirroring sessions are supported per standalone chassis-based switch or in a stack consisting of two or more switches.
- You cannot configure a port mirroring and a port monitoring session on the same NI module in an OmniSwitch chassis-based switch.
- You cannot configure port mirroring and monitoring on the same switching ASIC. Each switching ASIC controls 24 ports (e.g., ports 1–24, 25–48, etc.). For example, if a port mirroring session is configured for ports 1/12 and 1/22, then configuring a port monitoring session for any of the ports between 1 and 24 is not allowed.
- If a port mirroring session is configured across two switching ASICs, then configuring a monitoring session is not allowed on any of the ports controlled by each of the ASICs involved. For example, if a port mirroring session is configured for ports 1/8 and 1/30 on a 48-port switch, then configuring a port monitoring session involving any of the ports between 1 and 48 is not allowed.

What Ports Can Be Mirrored?

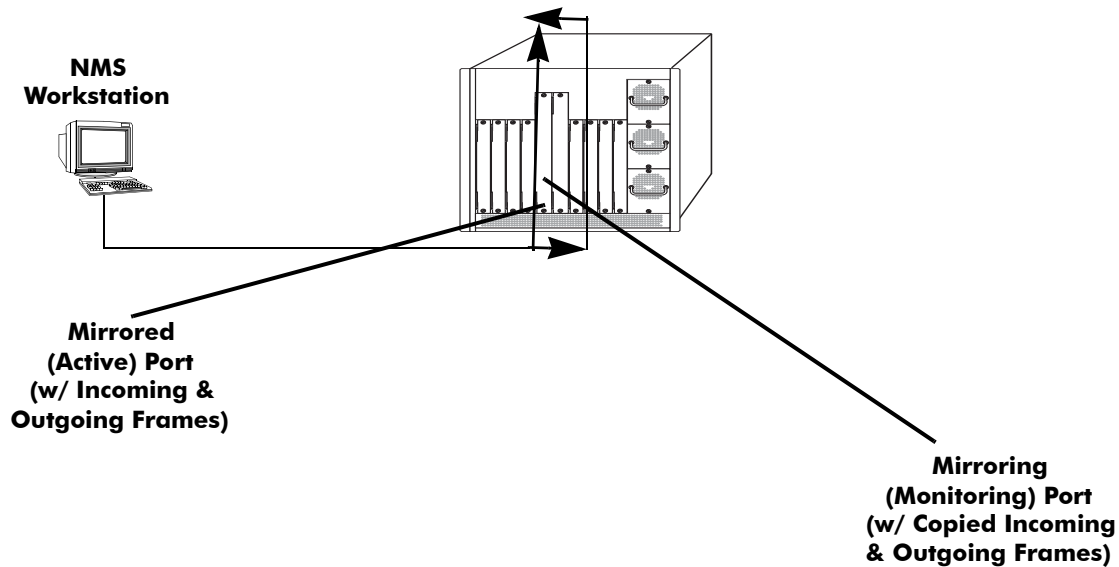
Mirroring between any 10/100/1000 port to any other 10/100/1000 port and between any SFP to any other SFP port is supported.

How Port Mirroring Works

When a frame is received on a mirrored port, it is copied and sent to the mirroring port. The received frame is actually transmitted twice across the switch backplane—once for normal bridging and then again to the mirroring port.

When a frame is transmitted by the mirrored port, a copy of the frame is made, tagged with the mirroring port as the destination, and sent back over the switch backplane to the mirroring port. The diagram below illustrates the data flow between the mirrored and mirroring ports.

Note that when port mirroring is enabled, there may be some performance degradation, since all frames received and transmitted by the mirrored port need to be copied and sent to the mirroring port.



Relationship Between Mirrored and Mirroring Ports

What Happens to the Mirroring Port

When you set up port mirroring and attach cables to the mirrored and mirroring ports, the mirroring port remains enabled and is a part of the Bridging Spanning Tree until you protect it from Spanning Tree updates by specifying an unblocked VLAN as part of the configuration command line. The mirroring port does not transmit or receive any traffic on its own.

Mirroring on Multiple Ports

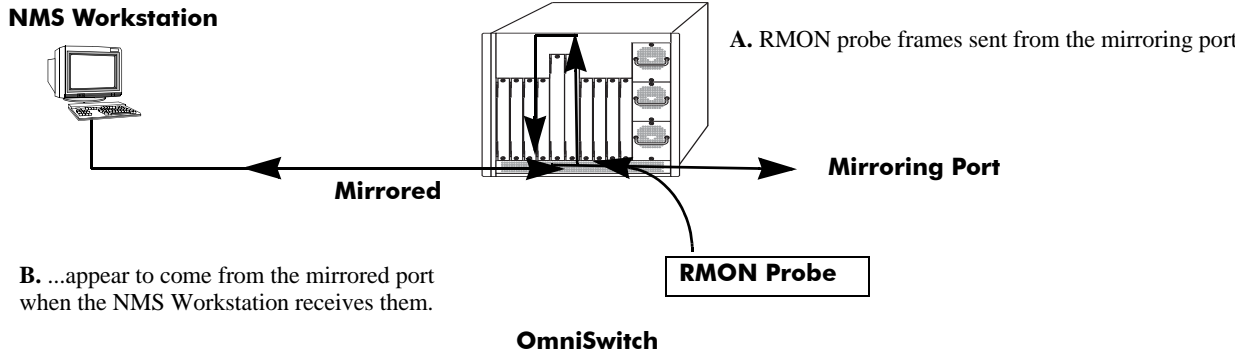
If mirroring is enabled on multiple ports and the same traffic is passing through these ports, then only one copy of each packet is sent to the mirroring destination. When the packet is mirrored for the first time, the switching ASIC flags the packet as “already mirrored.” If the packet goes through one more port where mirroring is enabled, that packet will not be mirrored again. If both mirroring and monitoring are enabled then the packet will be either mirrored or monitored (i.e., sent to CPU), whichever comes first.

Using Port Mirroring with External RMON Probes

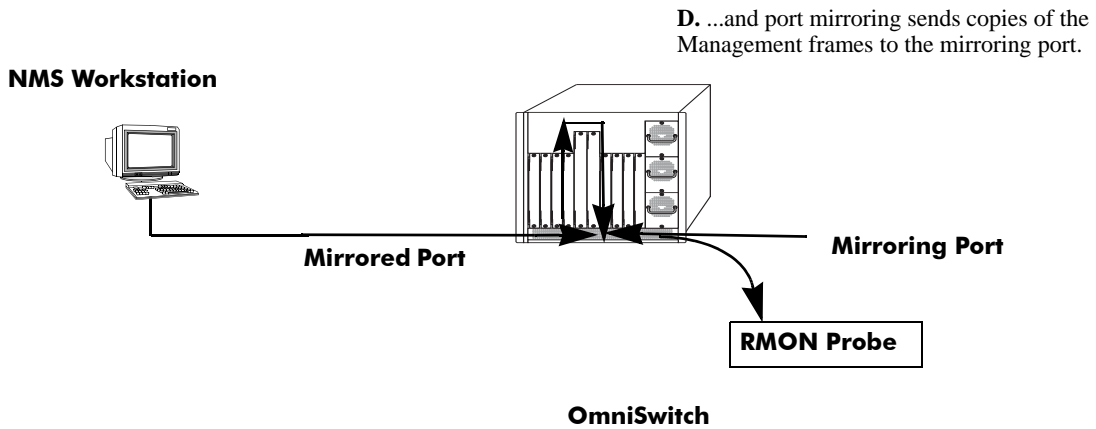
Port mirroring is a helpful monitoring tool when used in conjunction with an external RMON probe. Once you set up port mirroring, the probe can collect all relevant RMON statistics for traffic on the mirrored port. You can also move the mirrored port so that the mirroring port receives data from different ports. In this way, you can roam the switch and monitor traffic at various ports.

Note. If the mirroring port monitors mirrored traffic on an RMON probe belonging to a different VLAN than the mirrored port, it should be protected from blocking due to Spanning Tree updates. See [“Unblocking Ports \(Protection from Spanning Tree\)”](#) on page 38-19 for details.

The diagram on the following page illustrates how port mirroring can be used with an external RMON probe to copy RMON probe frames and Management frames to and from the mirroring and mirrored ports. Frames received from an RMON probe attached to the mirroring port can be seen as being received by the mirrored port. These frames from the mirroring port are marked as if they are received on the mirrored port before being sent over the switch backplane to an NMS station. Therefore, management frames destined for the RMON probe are first forwarded out of the mirrored port. After being received on the mirrored port, copies of the frames are mirrored out of the mirroring port—the probe attached to the mirroring port receives the management frames.



C. Management frames from the NMS Workstation are sent to the mirrored port....



Port Mirroring Using External RMON Probe

Remote Port Mirroring

Remote Port Mirroring expands the port mirroring functionality by allowing mirrored traffic to be carried over the network to a remote switch. With Remote Port Mirroring the traffic is carried over the network using a dedicated Remote Port Mirroring VLAN, no other traffic is allowed on this VLAN. The mirrored traffic from the source switch is tagged with the VLAN ID of the Remote Port Mirroring VLAN and forwarded over the intermediate switch ports to the destination switch where an analyzer is attached.

Since Remote Port Mirroring requires traffic to be carried over the network, the following exceptions to regular port mirroring exist:

- Spanning Tree must be disabled for the Remote Port Mirroring VLAN on all switches.
- There must not be any physical loop present in the Remote Port Mirroring VLAN.
- On the intermediate and destination switches, source learning must be disabled or overridden on the ports belonging to the Remote Port Mirroring VLAN.
- The QoS redirect feature can be used to override source learning on an OmniSwitch.

The following types of traffic will not be mirrored:

- Link Aggregation Control Packets (LACP)
- 802.1AB (LLDP)
- 802.1x port authentication
- 802.3ag (OAM)
- Layer 3 control packets
- Generic Attribute Registration Protocol (GARP)
- BPDUs.

For more information and an example of a Remote Port Mirroring configuration, see [“Remote Port Mirroring” on page 38-17](#).

Creating a Mirroring Session

Before port mirroring can be used, it is necessary to create a port mirroring session. The **port mirroring source destination** CLI command can be used to create a mirroring session between a mirrored (active) port and a mirroring port. Two (2) port mirroring sessions are supported in a standalone switch or in a stack consisting of two or more switches. In addition, “N-to-1” port mirroring is supported, where up to 24 source ports can be mirrored to a single destination port.

Note. To prevent the mirroring (destination) port from being blocked due to Spanning Tree changes, be sure to specify the VLAN ID number (from 1 to 4094) for the port that will remain **unblocked** (protected from these changes while port mirroring is active). This parameter is optional; if it is not specified, changes resulting from Spanning Tree could cause the port to become blocked (default). See **Unblocking Ports (Protection from Spanning Tree)** below for details.

To create a mirroring session, enter the **port mirroring source destination** command and include the port mirroring session ID number and the source and destination slot/ports, as shown in the following example:

```
-> port mirroring 6 source 2/3 destination 2/4
```

This command line specifies mirroring session 6, with the source (mirrored) port located in slot 2/port 3, and the destination (mirroring) port located in slot 3/port 4.

To create a remote port mirroring session, enter the **port mirroring source destination** command and include the port mirroring session ID number, the source and destination slot/ports, and the remote port mirroring VLAN ID as shown in the following example:

```
-> port mirroring 8 source 1/1 destination 1/2 rpmir-vlan 1000
```

This command line specifies remote port mirroring session 8, with the source (mirrored) port located on slot 1/port 1, the destination (mirroring) port on slot 1/port 2, and the remote port mirroring VLAN 1000.

Note. Neither the mirrored nor the mirroring ports can be a mobile port. See [Chapter 6, “Assigning Ports to VLANs,”](#) for information on mobile ports.

Creating an “N-to-1” port mirroring session is supported, where multiple source ports can be mirrored to a single destination port. In the following example, port 1/2, 2/1, and 2/3 are mirrored on destination port 2/4 in session 1:

```
-> port mirroring 1 source 1/2 destination 2/4
-> port mirroring 1 source 2/1 destination 2/4
-> port mirroring 1 source 2/3 destination 2/4
```

As an option, you can specify a range of source ports and/or multiple source ports. In the following example, ports 1/2 through 1/6 are mirrored on destination port 2/4 in session 1:

```
-> port mirroring 1 source 1/2-6 destination 2/4
```

In the following example, ports 1/9, 2/7, and 3/5 are mirrored on destination port 2/4 in session 1:

```
-> port mirroring 1 source 1/9 2/7 3/5 destination 2/4
```

In the following example, 1/2 through 1/6 and 1/9, 2/7, and 3/5 are mirrored on destination port 2/4 in session 1:

```
-> port mirroring 1 source 1/2-6 1/9 2/7 3/5 destination 2/4
```

Note. Ports can be added after a port mirroring session has been configured.

Unblocking Ports (Protection from Spanning Tree)

If the mirroring port monitors mirrored traffic on an RMON probe belonging to a different VLAN than the mirrored port, it should be protected from blocking due to Spanning Tree updates. To create a mirroring session that protects the mirroring port from being blocked (*default*) due to changes in Spanning Tree, enter the **port mirroring source destination** CLI command and include the port mirroring session ID number, source and destination slot/ports, and unblocked VLAN ID number, as shown in the following example:

```
-> port mirroring 6 source 2/3 destination 2/4 unblocked 750
```

This command line specifies mirroring session 6, with the source (mirrored) port located in slot 2/port 3, and the destination (mirroring) port located in slot 2/port 4. The mirroring port on VLAN 750 is protected from Spanning Tree updates.

Note. If the unblocked VLAN identifier is not specified, the mirroring port could be blocked due to changes in Spanning Tree.

Enabling or Disabling Mirroring Status

Mirroring Status is the parameter using which you can enable or disable a mirroring session (i.e., turn port mirroring on or off). There are two ways to do this:

- *Creating a Mirroring Session and Enabling Mirroring Status or Disabling a Mirroring Session (Disabling Mirroring Status).* These procedures are described below and on the following page.
- *Enabling or Disabling a Port Mirroring Session*—“shorthand” versions of the above commands that require fewer keystrokes. Only the port mirroring session ID number needs to be specified, rather than the entire original command line syntax (e.g., source and destination slot/ports and optional unblocked VLAN ID number). See [“Enabling or Disabling a Port Mirroring Session \(Shorthand\)”](#) on page 38-20 for details.

Disabling a Mirroring Session (Disabling Mirroring Status)

To disable the mirroring status of the configured session between a mirrored port and a mirroring port (turning port mirroring off), use the **port mirroring source destination** CLI command. Be sure to include the port mirroring session ID number and the keyword **disable**.

In this example, the command specifies port mirroring session 6, with the mirrored (active) port located in slot 2/port 3, and the mirroring port located in slot 6/port 4. The mirroring status is disabled (i.e., port mirroring is turned off):

```
-> port mirroring 6 source disable
```

Note. You can modify the parameters of a port mirroring session that has been disabled.

Keep in mind that the port mirroring session configuration remains valid, even though port mirroring has been turned off. Note that the port mirroring session identifier and slot/port locations of the designated interfaces must always be specified.

Note. Note that the port mirroring session identifier and slot/port locations of the designated interfaces must always be specified.

Configuring Port Mirroring Direction

By default, port mirroring sessions are bidirectional. To configure the direction of a port mirroring session between a mirrored port and a mirroring port, use the **port mirroring source destination** CLI command by entering port mirroring, followed by the port mirroring session ID number, the source and destination slot/ports, and **bidirectional**, **inport**, or **outport**.

Note. Optionally, you can also specify the optional unblocked VLAN ID number and either **enable** or **disable** on the same command line.

In this example, the command specifies port mirroring session 6, with the mirrored (active) port located in slot 2/port 3 and the mirroring port located in slot 6/port 4. The mirroring direction is unidirectional and inward bound:

```
-> port mirroring 6 source 2/3 destination 6/4 inport
```

In this example, the command specifies port mirroring session 6, with the mirrored (active) port located in slot 2/port 3, and the mirroring port located in slot 6/port 4. The mirroring direction is unidirectional and outward bound:

```
-> port mirroring 6 source 2/3 destination 6/4 outport
```

You can use the bidirectional keyword to restore a mirroring session to its default bidirectional configuration. For example:

```
-> port mirroring 6 source 2/3 destination 6/4 bidirectional
```

Note. Note that the port mirroring session identifier and slot/port locations of the designated interfaces must always be specified.

Enabling or Disabling a Port Mirroring Session (Shorthand)

Once a port mirroring session configuration has been created, this command is useful for enabling or disabling it (turning port mirroring on or off) without having to re-enter the source and destination ports and unblocked VLAN ID command line parameters.

To enable a port mirroring session, enter the **port mirroring** command, followed by the port mirroring session ID number and the keyword **enable**. The following command enables port mirroring session 6 (turning port mirroring on):

```
-> port mirroring 6 enable
```

Note. Port mirroring session parameters cannot be modified when a mirroring session is enabled. Before you can modify parameters, the mirroring session must be disabled.

To disable a port mirroring session, enter the **port mirroring** command, followed by the port mirroring session ID number and the keyword **disable**. The following command disables port mirroring session 6 (turning port mirroring off):

```
-> port mirroring 6 disable
```

Displaying Port Mirroring Status

To display port mirroring status, use the **show port mirroring status** command. To display all port mirroring sessions, enter:

```
-> show port mirroring status 6
```

Session	Mirror Destination	Mirror Direction	Unblocked Vlan	Config Status	Oper Status
1.	2/1	-	NONE	Enable	On
	Mirror Source				
1.	1/1	bidirectional	-	Enable	On
1.	1/2	bidirectional	-	Enable	On
1.	1/3	bidirectional	-	Enable	On
1.	1/4	bidirectional	-	Enable	On
1.	1/5	bidirectional	-	Enable	On

Deleting A Mirroring Session

The **no** form of the **port mirroring** command can be used to delete a previously created mirroring session configuration between a mirrored port and a mirroring port.

To delete a mirroring session, enter the **no port mirroring** command, followed by the port mirroring session ID number. For example:

```
-> no port mirroring 6
```

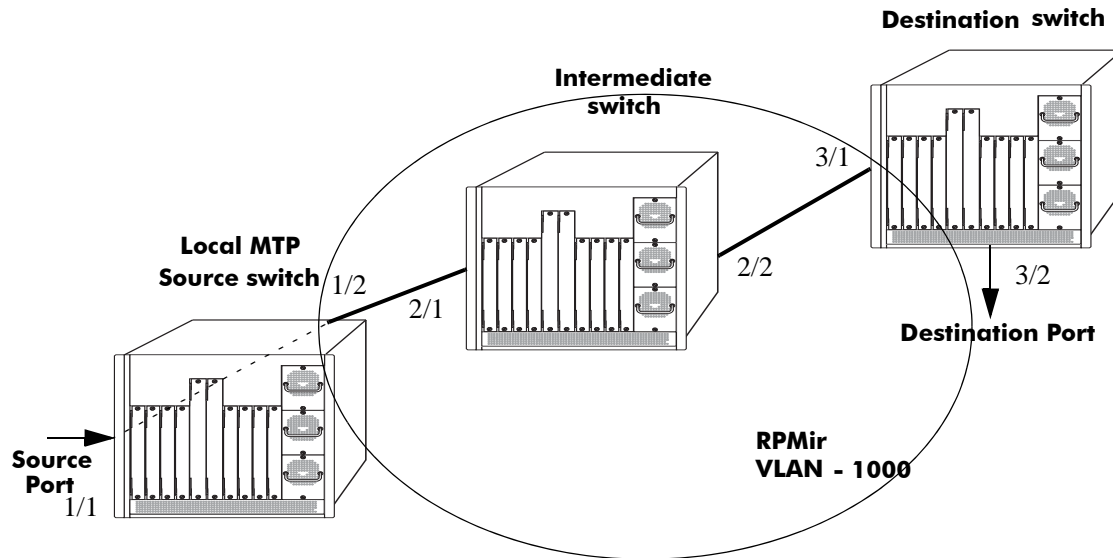
In this example, port mirroring session 6 is deleted.

Note. The port mirroring session identifier must always be specified.

Configuring Remote Port Mirroring

This section describes the steps required to configure Remote Port Mirroring between Source, Intermediate, and Destination switches.

The following diagram shows an example of a Remote Port Mirroring configuration:



Remote Port Mirroring Example

Configuring Source Switch

Follow the steps given below to configure the Source Switch:

- > vlan 1000
- > vlan 1000 stp disable
- > port mirroring 8 source 1/1
- > port mirroring 8 destination 1/2 rpmir-vlan 1000

Configuring Intermediate Switch

Follow the steps given below to configure all the Intermediate Switches:

- > vlan 1000
- > vlan 1000 stp disable
- > vlan 1000 802.1q 2/1
- > vlan 1000 802.1q 2/2

Enter the following QoS commands to override source learning:

- > policy condition c_is1 source vlan 1000
- > policy action a_is1 redirect port 2/2

-> policy rule r_is1 condition c_is1 action a_is1

-> qos apply

Note. If the intermediate switches are not OmniSwitches, refer to the vendor's documentation for instructions on disabling or overriding source learning.

Configuring Destination Switch

Follow the steps given below to configure the Destination Switch:

-> vlan 1000

-> vlan 1000 stp disable

-> vlan 1000 802.1q 3/1

-> vlan 1000 port default 3/2

Enter the following QoS commands to override source learning:

-> policy condition c_ds1 source vlan 1000

-> policy action a_ds1 redirect port 3/2

-> policy rule r_ds1 condition c_ds1 action a_ds1

-> qos apply

Port Monitoring

An essential tool of the network engineer is a network packet capture device. A packet capture device is usually a PC-based computer, such as the Sniffer[®], that provides a means for understanding and measuring data traffic of a network. Understanding data flow in a VLAN-based switch presents unique challenges, primarily because traffic moves inside the switch, especially on dedicated devices.

The port monitoring feature allows you to examine packets to and from a specific Ethernet port. Port monitoring has the following features:

- Software commands to enable and display captured port data.
- Captures data in Network General[®] file format.
- A file called **pmonitor.enc** is created in the **/flash** memory when you configure and enable a port monitoring session.
- Data packets time stamped.
- One port monitored at a time.
- RAM-based file system.
- Statistics gathering and display.

The port monitoring feature also has the following restrictions:

- All packets cannot be captured. (Estimated packet capture rate is around 500 packets/second.)
- The maximum number of monitoring sessions is limited to one per chassis and/or stack.
- You cannot configure a port mirroring and a port monitoring session on the same NI module in an OmniSwitch chassis-based switch.
- You cannot configure port mirroring and monitoring on the same switching ASIC. Each switching ASIC controls 24 ports (e.g., ports 1–24, 25–48, etc.). For example, if a port mirroring session is configured for ports 1/12 and 1/22, then configuring a port monitoring session for any of the ports between 1 and 24 is not allowed.
- If a port mirroring session is configured across two switching ASICs, then configuring a monitoring session is not allowed on any of the ports controlled by each of the ASICs involved. For example, if a port mirroring session is configured for ports 1/8 and 1/30 on a 48-port switch, then configuring a port monitoring session involving any of the ports between 1 and 48 is not allowed.
- Only the first 64 bytes of the traffic will be captured.
- Link Aggregation ports can be monitored.
- If both mirroring and monitoring are enabled, then packets will either be mirrored *or* monitored (i.e., sent to CPU), whichever comes first. See [“Mirroring on Multiple Ports” on page 38-15](#) for more information.

You can select to dump real-time packets to a file. Once a file is captured, you can FTP it to a Sniffer or PC for viewing.

Configuring a Port Monitoring Session

To configure a port monitoring session, use the **port monitoring source** command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), and the port number of the port.

For example, to configure port monitoring session 6 on port 2/3 enter:

```
-> port monitoring 6 source 2/3
```

Note. One port monitoring session can be configured per chassis or stack.

In addition, you can also specify optional parameters shown in the table below. These parameters must be entered after the slot and port number.

keywords

file	no file	size
no overwrite	inport	outport
bidirectional	timeout	enable
disable		

For example, to configure port monitoring session 6 on port 2/3 and administratively enable it, enter:

```
-> port monitoring 6 source 2/3 enable
```

These keywords can be used when creating the port monitoring session or afterwards. See the sections below for more information on using these keywords.

Enabling a Port Monitoring Session

To disable a port monitoring session, use the **port monitoring source** command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, and **enable**. For example, to enable port monitoring session 6 on port 2/3, enter:

```
-> port monitoring 6 source 2/3 enable
```

Disabling a Port Monitoring Session

To disable a port monitoring session, use the **port monitoring** command by entering **port monitoring**, followed by the port monitoring session ID and **pause**. For example, to disable port monitoring session 6, enter:

```
-> port monitoring 6 disable
```

Deleting a Port Monitoring Session

To delete a port monitoring session, use the **no** form of the **port monitoring** command by entering **no port monitoring**, followed by the port monitoring session ID. For example, to delete port monitoring session 6, enter:

```
-> no port monitoring 6
```

Pausing a Port Monitoring Session

To pause a port monitoring session, use the **port monitoring** command by entering **port monitoring**, followed by the port monitoring session ID and **pause**. For example, to pause port monitoring session 6, enter:

```
-> port monitoring 6 pause
```

To resume a paused port monitoring session, use the **port monitoring** command by entering **port monitoring**, followed by the port monitoring session ID and **resume**. For example, to resume port monitoring session 6, enter:

```
-> port monitoring 6 resume
```

Configuring Port Monitoring Session Persistence

By default, a port monitoring session will never be disabled. To modify the length of time before a port monitoring session is disabled from 0 (the default, where the session is permanent) to 2147483647 seconds, use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, **timeout**, and the number of seconds before it is disabled.

For example, to configure port monitoring session 6 on port 2/3 that will last 12000 seconds before it is disabled, enter:

```
-> port monitoring 6 source 2/3 timeout 12000
```

Configuring a Port Monitoring Data File

By default, a file called **pmonitor.enc** is created in the **/flash** directory when you configure and enable a port monitoring session. This file can be FTPed for later analysis. To configure a user-specified file, use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, **file**, and the name of the file.

For example, to configure port monitoring session 6 on port 2/3 with a data file called “user_port” in the **/flash** directory, enter:

```
-> port monitoring 6 source 2/3 file /flash/user_port
```

Optionally, you can also configure the size of the file and/or you can configure the data file so that more-recent packets will not overwrite older packets in the data file if the file size is exceeded.

To create a file and configure its size, use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, **file**, the name of the file, **size**, and the size of the file in 16K byte increments. (The maximum size is 140K bytes.)

For example, to configure port monitoring session 6 on port 2/3 with a data file called “user_port” in the **/flash** directory with a size of 49152 (3 * 16K), enter:

```
-> port monitoring 6 source 2/3 file /flash/user_port size 3
```

To prevent more recent packets from overwriting older packets in the data file, if the file size is exceeded, use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, **file**, the name of the file, and **overwrite off**.

For example, to configure port monitoring session 6 on port 2/3 with a data file called “user_port” in the **/flash** directory that will not overwrite older packets if the file size is exceeded, enter:

```
-> port monitoring 6 source 2/3 file user_port overwrite off
```

To allow more recent packets from overwriting older packets in the data file if the file size is exceeded (the default), use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, **file**, the name of the file, and **overwrite on**.

For example, to configure port monitoring session 6 on port 2/3 with a data file called “user_port” in the **/flash** directory that will not overwrite older packets if the file size is exceeded, enter:

```
-> port monitoring 6 source 2/3 file /flash/user_port overwrite on
```

Note. The **size** and **no overwrite** options can be entered on the same command line.

Suppressing Port Monitoring File Creation

By default, a file called **pmonitor.enc** is created in **/flash** memory when you configure and enable a port monitoring session. To prevent the file from being created, use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, and **no file**.

For example, to configure port monitoring session 6 on port 2/3 with no data file created enter:

```
-> port monitoring 6 source 2/3 no file
```

Configuring Port Monitoring Direction

By default, port monitoring sessions are bidirectional. To configure the direction of a port mirroring session between a mirrored port and a mirroring port, use the **port monitoring source** CLI command by entering **port monitoring**, followed by the user-specified session ID number, **source**, the slot number of the port to be monitored, a slash (/), the port number of the port, and **inport**, **outport**, or **bidirectional**.

For example, to configure port monitoring session 6 on port 2/3 as unidirectional and inward bound, enter:

```
-> port monitoring 6 source 2/3 inport
```

To configure port monitoring session 6 on port 2/3 as unidirectional and outward bound, for example, enter:

```
-> port monitoring 6 source 2/3 outport
```

For example, to restore port monitoring session 6 on port 2/3 to its bidirectional direction, enter:

```
-> port monitoring 6 source 2/3 bidirectional
```

Displaying Port Monitoring Status and Data

A summary of the show commands used for displaying port monitoring status and port monitoring data is given here:

show port monitoring status Displays port monitoring status.

show port monitoring file Displays port monitoring data.

For example, to display port monitoring data, use the **show port monitoring file** command as shown below:

```
-> show port monitoring file
```

Destination	Source	Type	Data
01:80:C2:00:00:00	00:20:DA:8F:92:C6	BPDU	00:26:42:42:03:00:00:00:00:00
00:20:DA:C7:2D:D6	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:FE:4A:40:00
00:20:DA:A3:89:F6	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:CF:89:40:00
00:20:DA:BF:5B:76	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:CF:85:40:00
00:20:DA:A3:89:F6	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:CF:8A:40:00
00:20:DA:BF:5B:76	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:CF:86:40:00
00:20:DA:A3:89:F6	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:CF:8B:40:00
01:80:C2:00:00:00	00:20:DA:8F:92:C6	BPDU	00:26:42:42:03:00:00:00:00:00
00:20:DA:BF:5B:76	08:00:20:95:F3:89	UDP	08:00:45:00:00:6B:CF:87:40:00

Note. For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

sFlow

sFlow is a network monitoring technology that gives visibility in to the activity of the network, by providing network usage information. It provides the data required to effectively control and manage the network usage. sFlow is a sampling technology that meets the requirements for a network traffic monitoring solution.

sFlow is an industry standard with many vendors delivering products with this support. Some of the applications of the sFlow data include:

- Detecting, diagnosing, and fixing network problems
- Real-time congestion management
- Detecting unauthorized network activity
- Usage accounting and billing
- Understanding application mix
- Route profiling and peer optimization
- Capacity planning

sFlow is a sampling technology embedded within switches/routers. It provides the ability to monitor the traffic flows. It requires a sFlow agent software process running as part of the switch software and a sFlow collector which receives and analyses the monitored data. The sFlow collector makes use of SNMP to communicate with a sFlow agent in order to configure sFlow monitoring on the device (switch).

sFlow agent running on the switch/router, combines interface counters and traffic flow (packet) samples preferably on all the interfaces into sFlow datagrams that are sent across the network to a sFlow collector.

Packet sampling on the switch/router is typically performed by the switching/routing ASICs, providing wire-speed performance. In this case, sFlow agent does very little processing, by packaging data into sFlow datagrams that are immediately sent on network. This minimizes the memory and CPU utilization by sFlow agent.

sFlow Manager

The sFlow manager is the controller for all the modules. It initializes all other modules. It interfaces with the Ethernet driver to get the counter samples periodically and reads sampled packets from the Q-Dispatcher module. The counter samples are given to the poller module and sampled packets are given to the sampler to format a UDP. The sFlow manager also has a timer which periodically sends timer ticks to other sections.

Each sFlow manager instance has multiples of receiver, sampler, and poller instances. Each user programmed port will have an individual sampler and poller. The sampler and poller could be potentially pointing to multiple receivers if the user has configured multiple destination hosts.

Receiver

The receiver module has the details about the destination hosts where the sFlow datagrams are sent out. If there are multiple destination then each destination has an instance of the receiver. All these receivers are attached to the sFlow manager instance and to an associated sample/poller.

Sampler

The sampler is the module which gets hardware sampled from Q-Dispatcher and fills up the sampler part of the UDP datagram.

Poller

The poller is the module which gets counter samples from Ethernet driver and fills up the counter part of the UDP datagram.

Configuring a sFlow Session

To configure a sFlow receiver session, use the **sflow receiver** command by entering **sflow receiver**, followed by the receiver_index, name, the name of the session and **address**, and the IP address of the switch to be monitored.

For example, to configure receiver session 6 on switch 10.255.11.28, enter:

```
-> sflow receiver 6 name sflowtrend address 10.255.11.28
```

In addition, you can also specify optional parameters shown in the table below. These parameters can be entered after the IP address.

keywords

timeout	packet-size
forever	version
udp-port	

For example, to configure sFlow receiver session 6 on switch 10.255.11.28 and to specify the packet-size and timeout value, enter:

```
-> sflow receiver 6 name sflowtrend address 10.255.11.28 packet-size 1400 time-out 600
```

To configure a sFlow sampler session, use the **sflow sampler** command by entering **sflow sampler**, followed by the instance ID number, the slot number of the port to be monitored, a slash (/), and the port number and **receiver**, the receiver_index.

For example, to configure sampler session 1 on port 2/3, enter:

```
-> sflow sampler 1 2/3 receiver 6
```

In addition, you can also specify optional parameters shown in the table below. These parameters can be entered after the receiver index.

keywords

rate
sample-hdr-size

For example, to configure sFlow sampler session 1 on port 2/3 and to specify the rate and sample-hdr-size, enter:

```
-> sflow sampler 1 2/3 receiver 6 rate 512 sample-hdr-size 128
```

To configure a sFlow poller session, use the **sflow poller** command by entering **sflow poller**, followed by the instance ID number, the slot number of the port to be monitored, a slash (/), and the port number of the port and **receiver**, then *receiver_index*.

For example, to configure poller session 3 on port 1/1, enter:

```
-> sflow poller 3 1/1 receiver 6
```

In addition, you can also specify the optional **interval** parameter after the receiver index value. For example, to configure sFlow poller session 3 on port 1/1 with an interval of 5, enter:

```
-> sflow poller 3 1/1 receiver 6 interval 5
```

Configuring a Fixed Primary Address

It is necessary to execute the **ip interface** command to make a loopback0 IP address as the fixed primary address of the switch, in order to avoid interface changes, which might need the collector software to be restarted for it to communicate using the new agent IP address. Normally, the primary IP address could change depending on the IP interface going up/down. Therefore, the sFlow agent always needs to send a fixed IP address in the datagram.

For example, to configure the loopback0 address as a primary IP address, enter:

```
-> ip interface loopback0 address 198.206.181.100
```

Note. The loopback address should be an IP interface configured on the switch.

Displaying a sFlow Receiver

The **show sflow receiver** command is used to display the receiver table.

For example, to view the sFlow receiver table, enter the **show sflow receiver** command without specifying any additional parameters. A screen similar to the following example will be displayed, as shown below:

```
-> show sflow receiver

Receiver 1
Name      = Golden
Address   = IP_V4 198.206.181.3
UDP Port  = 6343
Timeout   = 65535
Packet Size= 1400
DatagramVer= 5
```

Note. For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Displaying a sFlow Sampler

The **show sflow sampler** command is used to display the sampler table.

For example, to view the sFlow sampler table, enter the **show sflow sampler** command without specifying any additional parameters. A screen similar to the following example will be displayed, as shown below:

```
-> show sflow sampler
```

Instance	Interface	Receiver	Sample-rate	Sample-hdr-size
1	2/ 1	1	2048	128
1	2/ 2	1	2048	128
1	2/ 3	1	2048	128
1	2/ 4	1	2048	128
1	2/ 5	1	2048	128

Note. For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Displaying a sFlow Poller

The **show sflow poller** command is used to display the poller table.

For example, to view the sFlow poller table, enter the **show sflow poller** command without specifying any additional parameters. A screen similar to the following example will be displayed, as shown below:

```
-> show sflow poller
```

Instance	Interface	Receiver	Interval
1	2/ 6	1	30
1	2/ 7	1	30
1	2/ 8	1	30
1	2/ 9	1	30
1	2/10	1	30

Note. For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Displaying a sFlow Agent

The **show sflow agent** command is used to display the receiver table.

For example, to view the sFlow agent table, enter the **show sflow agent** command without specifying any additional parameters. A screen similar to the following example will be displayed, as shown below:

```
-> ip interface loopback0 127.0.0.1
-> show sflow agent

Agent Version   = 1.3; Alcatel-Lucent; 6.1.1
Agent IP        = 127.0.0.1
```

Note. For more information about the displays that result from these commands, see the *OmniSwitch 6250 CLI Reference Guide*.

Deleting a sFlow Session

To delete a sFlow receiver session, use the release form at the end of the **sflow receiver** command by entering **sflow receiver**, followed by the receiver index and **release**. For example, to delete sFlow receiver session 6, enter:

```
-> sflow receiver 6 release
```

To delete a sFlow sampler session, use the no form of the **sflow sampler** command by entering **no sflow sampler**, followed by the instance ID number, the slot number of the port to delete, a slash (/), and the port number of the port, enter:

```
-> no sflow sampler 1 2/3
```

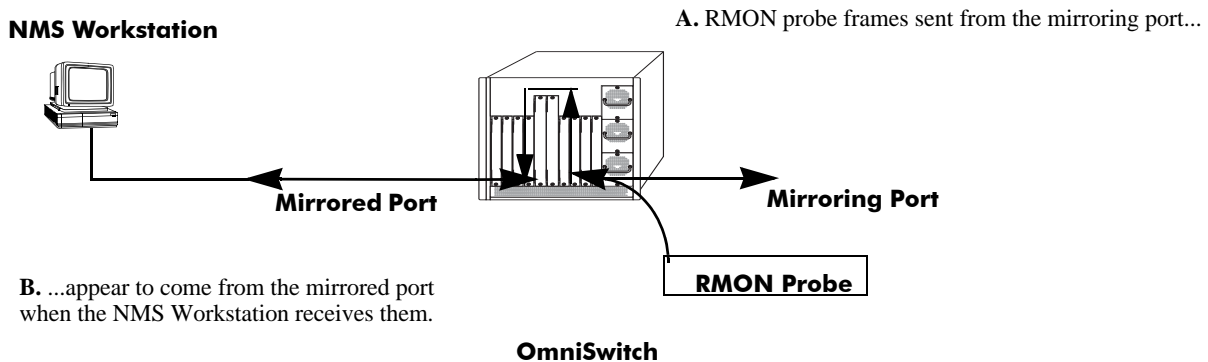
To delete a sFlow poller session, use the no form of the **sflow poller** command by entering **no sflow poller**, followed by the instance ID number, the slot number of the port to delete, a slash (/), and the port number of the port, enter:

```
-> no sflow poller 3 1/1
```

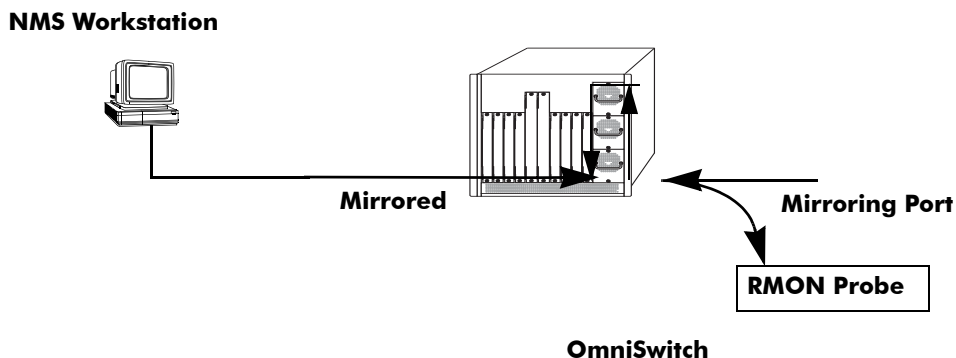
Remote Monitoring (RMON)

Remote Network Monitoring (RMON) is an SNMP protocol used to manage networks remotely. *RMON probes* can be used to collect, interpret, and forward statistical data about network traffic from designated active ports in a LAN segment to an NMS (Network Management System) application for monitoring and analysis without negatively impacting network performance. RMON software is fully integrated in the Chassis Management software and works with the Ethernet software to acquire statistical information. However, it does not monitor the CMM module's onboard Ethernet Management port on OmniSwitch chassis-based switches (which is reserved for management purposes).

The following diagram illustrates how an External RMON probe can be used with port mirroring to copy RMON probe frames and Management frames to and from the mirroring and mirrored ports. Frames received from an RMON probe attached to the mirroring port can be seen as being received by the mirrored port. These frames from the mirroring port are marked as if they are received on the mirrored port before being sent over the switch backplane to an NMS station. Therefore, management frames that are destined for the RMON probe are first forwarded out of the mirrored port. After being received on the mirrored port, copies of the frames are mirrored out of the mirroring port—the probe attached to the mirroring port receives the management frames.



C. Management frames from the NMS Workstation are sent to the mirrored port...



D. ...and port mirroring sends copies of the Management frames to the mirroring port.

Port Mirroring Using External RMON Probe

RMON probes can be enabled or disabled via CLI commands. Configuration of Alarm threshold values for RMON traps is a function reserved for RMON-monitoring NMS stations.

This feature supports basic RMON 4 group implementation in compliance with RFC 2819, including the **Ethernet Statistics**, **History** (Control & Statistics), **Alarms** and **Events** groups (*described below*).

Note. RMON 10 group and RMON2 are not implemented in the current release. An external RMON probe that includes RMON 10 group and RMON2 may be used where full RMON probe functionality is required.

Ethernet Statistics

Ethernet statistics probes are created whenever new ports are inserted and activated in the chassis. When a port is removed from the chassis or deactivated, the Ethernet statistics group entry associated with the physical port is invalidated and the probe is deleted.

The Ethernet statistics group includes port utilization and error statistics measured by the RMON probe for each monitored Ethernet interface on the switch. Examples of these statistics include CRC (Cyclic Redundancy Check)/alignment, undersized/oversized packets, fragments, broadcast/multicast/unicast, and bandwidth utilization statistics.

History (Control & Statistics)

The History (Control & Statistics) group controls and stores periodic statistical samplings of data from various types of networks. Examples include Utilization, Error Count, and Frame Count statistics.

Alarm

The Alarm group collects periodic statistical samples from variables in the probe and compares them to previously configured thresholds. If a sample crosses a previously configured threshold value, an Event is generated. Examples include Absolute or Relative Values, Rising or Falling Thresholds on the Utilization Frame Count and CRC Errors.

Event

The Event group controls generation and notification of events from the switch to NMS stations. For example, customized reports based on the type of Alarm can be generated, printed and/or logged.

Note. The following RMON groups are not implemented: **Host**, **HostTopN**, **Matrix**, **Filter**, and **Packet Capture**.

Enabling or Disabling RMON Probes

To enable or disable an individual RMON probe, enter the **rmon probes** CLI command. Be sure to specify the type of probe (**stats/history/alarm**), followed by the entry number (optional), as shown in the following examples.

The following command enables RMON Ethernet Statistics probe number 4012:

```
-> rmon probes stats 4012 enable
```

The following command disables RMON History probe number 10240:

```
-> rmon probes history 10240 disable
```

The following command enables RMON Alarm probe number 11235:

```
-> rmon probes alarm 11235 enable
```

To enable or disable an entire group of RMON probes of a particular flavor type (such as Ethernet Statistics, History, or Alarm), enter the command **without** specifying an *entry-number*, as shown in the following examples.

The following command disables all currently defined (disabled) RMON Ethernet Statistics probes:

```
-> rmon probes stats disable
```

The following command enables all currently defined (disabled) RMON History probes:

```
-> rmon probes history enable
```

The following command enables all currently defined (disabled) RMON Alarm probes:

```
-> rmon probes alarm enable
```

Note. Network activity on subnetworks attached to an RMON probe can be monitored by Network Management Software (NMS) applications.

Displaying RMON Tables

Two separate commands can be used to retrieve and view Remote Monitoring data: **show rmon probes** and **show rmon events**. The retrieved statistics appear in a *table* format (a collection of related data that meets the criteria specified in the command you entered). These RMON tables can display the following kinds of data (depending on the criteria you've specified):

- The **show rmon probes** command can display a list of current RMON probes or statistics for a particular RMON probe.
- The **show rmon events** command can display a list of RMON events (actions that occur in response to Alarm conditions detected by an RMON probe) or statistics for a particular RMON event.

Displaying a List of RMON Probes

To view a list of current RMON probes, enter the **show rmon probes** command with the probe type, without specifying an entry number for a particular probe.

For example, to show a list of the statistics probes, enter:

```
-> show rmon probes stats
```

A display showing all current statistics RMON probes should appear, as shown in the following example:

Entry	Slot/Port	Flavor	Status	Duration	System Resources
4001	4/1	Ethernet	Active	00:25:00	275 bytes
4008	4/8	Ethernet	Active	00:25:00	275 bytes
4005	4/5	Ethernet	Active	00:25:00	275 bytes

This table entry displays probe statistics for all probes on the switch. The probes are active, utilize 275 bytes of memory, and 25 minutes have elapsed since the last change in status occurred.

To show a list of the history probes, enter:

```
-> show rmon probes history
```

A display showing all current history RMON probes should appear, as shown in the following example:

Entry	Slot/Port	Flavor	Status	Duration	System Resources
1	1/1	History	Active	92:52:20	5464 bytes
30562	1/35	History	Active	00:31:22	312236 bytes
30817	1/47	History	Active	00:07:31	5200236 bytes

The table entry displays statistics for RMON History probes on the switch.

To show a list of the alarm probes, enter:

```
-> show rmon probes alarm
```

A display showing all current alarm RMON probes should appear, as shown in the following example:

Entry	Slot/Port	Flavor	Status	Duration	System Resources
31927	1/35	Alarm	Active	00:25:51	608 bytes

Displaying Statistics for a Particular RMON Probe

To view statistics for a particular current RMON probe, enter the `show rmon probes` command, specifying an entry number for a particular probe, such as:

```
-> show rmon probes 4005
```

A display showing statistics for the specified RMON probe will appear, as shown in the following sections.

Sample Display for Ethernet Statistics Probe

The display shown here identifies RMON Probe 4005's Owner description and interface location (OmniSwitch Auto Probe on slot 4, port 5), Entry number (4005), probe Flavor (Ethernet statistics), and Status (Active). Additionally, the display indicates the amount of time that has elapsed since the last change in status (48 hours, 54 minutes), and the amount of memory allocated to the probe, measured in bytes (275).

```
-> show rmon probes 4005
```

```
Probe's Owner: Switch Auto Probe on Slot 4, Port 5
Entry 4005
Flavor = Ethernet, Status = Active
Time = 48 hrs 54 mins,

System Resources (bytes) = 275
```

Sample Display for History Probe

The display shown here identifies RMON Probe 10325's Owner description and interface location (Analyzer-p:128.251.18.166 on slot 1, port 35), the total number of History Control Buckets (samples) requested and granted (2), along with the time interval for each sample (30 seconds) and system-generated Sample Index ID number (5859). The probe Entry number identifier (10325), probe Flavor (History), and Status (Active), the amount of time that has elapsed since the last change in status (48 hours, 53 minutes), and the amount of memory allocated to the probe, measured in bytes (601) are also displayed.

```
-> show rmon probes history 30562

Probe's Owner: Analyzer-p:128.251.18.166 on Slot 1, Port 35

History Control Buckets Requested    = 2
History Control Buckets Granted      = 2
History Control Interval              = 30 seconds
History Sample Index                 = 5859
Entry 10325
    Flavor = History, Status = Active
    Time = 48 hrs 53 mins,
    System Resources (bytes) = 601
```

Sample Display for Alarm Probe

The display shown here identifies RMON Probe 11235's Owner description and interface location (Analyzer-t:128.251.18.166 on slot 1, port 35), as well as the probe's Alarm Rising Threshold and Alarm Falling Threshold, maximum allowable values beyond which an alarm will be generated and sent to the Event group (5 and 0, respectively).

Additionally, the corresponding Alarm Rising Event Index number (26020) and Alarm Falling Event Index number (0), which link the Rising Threshold Alarm and Falling Threshold Alarm to events in the Event table, are identified. The Alarm Interval, a time period during which data is sampled (10 seconds) and Alarm Sample Type (delta value—variable) are also shown, as is the Alarm Variable ID number (1.3.6.1.2.1.16.1.1.1.5.4008). The probe Entry number identifier (11235), probe Flavor (Alarm), Status (Active), the amount of time that has elapsed since the last change in status (48 hours, 48 minutes), and the amount of memory allocated to the probe, measured in bytes (1677) are also displayed.

```
-> show rmon probes alarm 31927

Probe's Owner: Analyzer-t:128.251.18.166 on Slot 1, Port 35
Alarm Rising Threshold                = 5
Alarm Falling Threshold                = 0
Alarm Rising Event Index               = 26020
Alarm Falling Event Index              = 0
Alarm Interval                        = 10 seconds
Alarm Sample Type                      = delta value
Alarm Startup Alarm                   = rising alarm
Alarm Variable                         = 1.3.6.1.2.1.16.1.1.1.5.4008
Entry 11235
    Flavor = Alarm, Status = Active
    Time = 48 hrs 48 mins,
    System Resources (bytes) = 1677
```

Displaying a List of RMON Events

RMON Events are actions that occur based on Alarm conditions detected by an RMON probe. To view a list of logged RMON Events, enter the **show rmon events** command without specifying an entry number for a particular probe, such as:

```
-> show rmon events
```

A display showing all logged RMON Events should appear, as shown in the following example:

Entry	Time	Description
1	00:08:00	etherStatsPkts.4008: [Falling trap] "Falling Event"
2	00:26:00	etherStatsCollisions.2008: "Rising Event"
3	00:39:00	etherStatsCollisions.2008: "Rising Event"

The display shown above identifies the Entry number of the specified Event, along with the elapsed time since the last change in status (measured in hours/minutes/seconds) and a description of the Alarm condition detected by the probe for all RMON Logged Events. For example, Entry number 3 is linked to etherStatsCollisions.2008: [Rising trap] "Rising Event," an Alarm condition detected by the RMON probe in which a trap was generated based on a Rising Threshold Alarm, with an elapsed time of 39 minutes since the last change in status.

Displaying a Specific RMON Event

To view information for a specific logged RMON Event, enter the **show rmon events** command, specifying an entry number (event number) for a particular probe, such as:

```
-> show rmon events 3
```

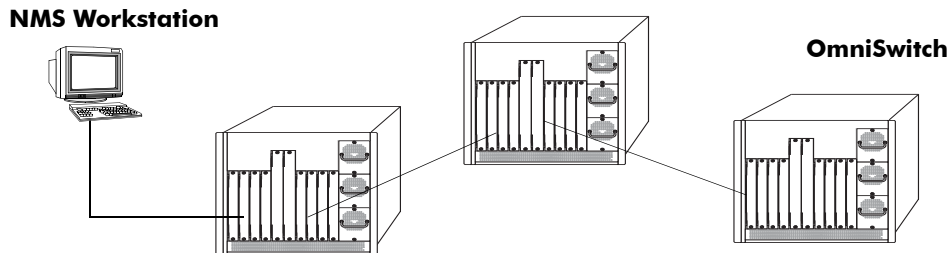
A display showing the specific logged RMON Event should appear, as shown in the following example:

Entry	Time	Description
3	00:39:00	etherStatsCollisions.2008: "Rising Event"

The display shown above identifies the Entry number of the specified Event, along with the elapsed time since the last change in status (measured in hours/minutes/seconds) and a description of the Alarm condition detected by the probe for the specific RMON Logged Event. For example, Entry number 3 is linked to etherStatsCollisions.2008: [Rising trap] "Rising Event," an Alarm condition detected by the RMON probe in which a trap was generated based on a Rising Threshold Alarm, with an elapsed time of 39 minutes since the last change in status.

Monitoring Switch Health

To monitor resource availability, the NMS (Network Management System) needs to collect significant amounts of data from each switch. As the number of ports per switch (and the number of switches) increases, the volume of data can become overwhelming. The Health Monitoring feature can identify and monitor a switch's resource utilization levels and thresholds, improving efficiency in data collection.



Monitoring Resource Availability from Multiple Ports and Switches

Health Monitoring provides the following data to the NMS:

- Switch-level Input/Output, Memory and CPU Utilization Levels
- Module-level and Port-level Input/Output Utilization Levels

For each monitored resource, the following variables are defined:

- Most recent utilization level (percentage)
- Average utilization level over the last minute (percentage)
- Average utilization level over the last hour (percentage)
- Maximum utilization level over the last hour (percentage)
- Threshold level

Additionally, Health Monitoring provides the capacity to specify thresholds for the resource utilization levels it monitors and generates traps based on the specified threshold criteria.

The following sections include a discussion of CLI commands that can be used to configure resource parameters and monitor or reset statistics for switch resources. These commands include:

- **health threshold**—Configures threshold limits for input traffic (RX), output/input traffic (TX/RX), memory usage, CPU usage, and chassis temperature. See [page 38-43](#) for more information.
- **show health threshold**—Displays current health threshold settings. See [page 38-44](#) for details.
- **health interval**—Configures sampling interval between health statistics checks. See [page 38-45](#) for more information.
- **show health interval**—Displays current health sampling interval, measured in seconds. See [page 38-45](#) for details.
- **show health** —Displays health statistics for the switch, as percentages of total resource capacity. See [page 38-46](#) for more information.
- **health statistics reset**—Resets health statistics for the switch. See [page 38-47](#) for details.

Configuring Resource and Temperature Thresholds

Health Monitoring software monitors threshold levels for the switch's consumable resources—*bandwidth, RAM memory, and CPU capacity*—as well as the ambient chassis temperature. When a threshold is exceeded, the Health Monitoring feature sends a trap to the Network Management Station (NMS). A trap is an alarm alerting the user to specific network events. In the case of health-related traps, a specific indication is given to determine which threshold has been crossed.

Note. When a resource falls back below the configured threshold, an addition trap is sent to the user. This indicates that the resource is no longer operating beyond its configured threshold limit.

The **health threshold** command is used to configure threshold limits for input traffic (RX), output/input traffic (TX/RX), memory usage, CPU usage and chassis temperature.

To configure thresholds for these resources, enter the **health threshold** command, followed by the input traffic, output/input traffic, memory usage, CPU usage, or chassis temperature value, where:

rx	Specifies an input traffic (RX) threshold, in percentage. This value defines the maximum percentage of total bandwidth allowed for <i>incoming traffic only</i> . The total bandwidth is the Ethernet port capacity of <i>all NI modules</i> currently operating in the switch, in Mbps. For example, a chassis with 48 100Base-T Ethernet ports installed has a total bandwidth of 4800 Mbps. Since the default RX threshold is 80 percent, the threshold is exceeded if the input traffic on all ports reaches 3840 Mbps or higher.
txrx	Specifies a value for the output/input traffic (TX/RX) threshold. This value defines the maximum percentage of total bandwidth allowed for <i>all incoming and outgoing traffic</i> . As with the RX threshold described above, the total bandwidth is defined as the Ethernet port capacity for all NI modules currently operating in the switch, in Mbps. The default TX/RX threshold is 80 percent.
memory	Specifies a value for the memory usage threshold. Memory usage refers to the total amount of RAM memory currently used by switch applications. The default memory usage threshold is 80 percent.
cpu	Specifies a value for the CPU usage threshold. CPU usage refers to the total amount of CPU processor capacity currently used by switch applications. The default CPU usage threshold is 80 percent.
temperature	Specifies a value for the chassis temperature threshold (Celsius). The default temperature threshold is 60 degrees Celsius.

For example, to specify a CPU usage threshold of 85 percent, enter the following command:

```
-> health threshold cpu 85
```

For more information on the **health threshold** command, refer to [Chapter 38, “Health Monitoring Commands,”](#) in the *OmniSwitch 6250 CLI Reference Guide*.

Note. When you specify a new value for a threshold limit, the value is automatically applied across all levels of the switch (switch, module, and port). You cannot select differing values for each level.

Displaying Health Threshold Limits

The **show health threshold** command is used to view all current health thresholds on the switch, as well as individual thresholds for input traffic (RX), output/input traffic (TX/RX), memory usage, CPU usage, and chassis temperature.

To view all health thresholds, enter the following command:

```
-> show health threshold
Rx Threshold           = 80
TxRx Threshold        = 80
Memory Threshold      = 80
CPU Threshold         = 80
Temperature Threshold = 60
```

To display a specific health threshold, enter the **show health threshold** command, followed by the appropriate suffix syntax:

- **rx**
- **txrx**
- **memory**
- **cpu**
- **temperature**

For example, if you want to view only the health threshold for memory usage, enter the following command:

```
-> show health threshold memory
Memory Threshold      = 80
```

Note. For detailed definitions of each of the threshold types, refer to [“Configuring Resource and Temperature Thresholds”](#) on page 38-43, as well as [Chapter 38, “Health Monitoring Commands,”](#) in the *OmniSwitch 6250 CLI Reference Guide*.

Configuring Sampling Intervals

The **sampling interval** is the period of time between polls of the switch's consumable resources to monitor performance vis-a-vis previously specified thresholds. The **health interval** command can be used to configure the sampling interval between health statistics checks.

To configure the sampling interval, enter the **health interval** command, followed by the number of seconds.

For example, to specify a **sampling interval** value of 6 seconds, enter the following command:

```
-> health interval 6
```

Valid values for the seconds parameter include 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, or 30.

Note. If the sampling interval is decreased, switch performance may be affected.

Viewing Sampling Intervals

The **show health interval** command can be used to display the current health sampling interval (period of time between health statistics checks), measured in seconds.

To view the sampling interval, enter the **show health interval** command. The currently configured health sampling interval (measured in seconds) will be displayed, as shown below:

```
-> show health interval
```

```
Sampling Interval = 5
```

Viewing Health Statistics for the Switch

The **show health** command can be used to display health statistics for the switch.

To display health statistics, enter the **show health** command, followed by the slot/port location and optional **statistics** keyword.

For example, to view health statistics for the entire switch, enter the **show health** command without specifying any additional parameters. A screen similar to the following example will be displayed, as shown below:

```
-> show health
* - current value exceeds threshold

Device          1 Min  1 Hr  1 Hr
Resources      Limit  Curr  Avg  Avg  Max
-----+-----+-----+-----+-----
Receive         80    00   00   00   00
Transmit/Receive 80    00   00   00   00
Memory          80    87*  87   86   87
Cpu             80    08   05   04   08
Temperature Cmm 60    34   34   33   34
Temperature Cmm Cpu 60    28   28   27   28
```

In the screen sample shown above, the Device Resources field displays the device resources that are being measured (for example, Receive displays statistics for traffic received by the switch; Transmit/Receive displays statistics for traffic transmitted and received by the switch; Memory displays statistics for switch memory; and CPU displays statistics for the switch CPU). The Limit field displays currently configured device threshold levels as percentages of available bandwidth. The Curr field displays current bandwidth usage for the specified device resource. 1 Min. Avg. refers to the average device bandwidth used over a 1 minute period. 1 Hr. Avg. refers to the average device bandwidth used over a 1 hour period, and 1 Hr. Max. refers to the maximum device bandwidth used over a 1 hour period.

Note. If the Current value appears with an asterisk displayed next to it, the Current value exceeds the Threshold limit. For example, if the Current value for Memory is displayed as 85* and the Threshold Limit is displayed as 80, the asterisk indicates that the Current value has exceeded the Threshold Limit value.

Viewing Health Statistics for a Specific Interface

To view health statistics for slot 4/port 3, enter the **show health** command, followed by the appropriate slot and port numbers. A screen similar to the following example will be displayed, as shown below:

```
-> show health 4/3
* - current value exceeds threshold

Port 04/03
Resources          Limit      Curr      1 Min      1 Hr      1 Hr
                  +-----+ +-----+ +-----+ +-----+ +-----+
                  Avg          Avg          Avg          Avg          Max
-----+-----+-----+-----+-----+
Receive            80        01        01         01         01
Transmit/Receive  80        01        01         01         01
```

In the screen sample shown above, the port 04/03 Resources field displays the port resources that are being measured (for example, Receive displays statistics for traffic received by the switch, while Transmit/Receive displays statistics for traffic transmitted and received by the switch). The Limit field displays currently configured resource threshold levels as percentages of available bandwidth. The Curr field displays current bandwidth usage for the specified resource. 1 Min. Avg. refers to the average resource bandwidth used over a 1 minute period. 1 Hr. Avg. refers to the average resource bandwidth used over a 1 hour period, and 1 Hr. Max. refers to the maximum resource bandwidth used over a 1 hour period.

Resetting Health Statistics for the Switch

The **health statistics reset** command can be used to clear health statistics for the entire switch. This command cannot be used to clear statistics only for a specific module or port.

To reset health statistics for the switch, enter the **health statistics reset** command, as shown below:

```
-> health statistics reset
```


39 Using Switch Logging

Switch logging is an event logging utility that is useful in maintaining and servicing the switch. Switch logging uses a formatted string mechanism to either record or discard event data from switch applications. The log records are copied to the output devices configured for the switch. Log records can be sent to a text file and written into the flash file system. The log records can also be scrolled to the switch's console or to a remote IP address.

Switch logging information can be customized and configured through Command Line Interface (CLI) commands, WebView, and SNMP. Log information can be helpful in resolving configuration or authentication issues, as well as general switch errors.

This chapter describes the switch logging feature, how to configure it and display switch logging information through the Command Line Interface (CLI). CLI commands are used in the configuration examples. For more details about the syntax of commands, see the *OmniSwitch 6250 CLI Reference Guide*.

In This Chapter

The following procedures are described:

- [“Enabling Switch Logging” on page 39-6](#)
- [“Setting the Switch Logging Severity Level” on page 39-6](#)
- [“Specifying the Switch Logging Output Device” on page 39-9](#)
- [“Displaying Switch Logging Status” on page 39-10](#)
- [“Displaying Switch Logging Records” on page 39-12](#)

Notes. Switch logging commands are not intended for use with low-level hardware and software debugging. It is strongly recommended that you contact an Alcatel-Lucent Customer Service representative for assistance with debugging functions.

Switch Logging Specifications

Platforms Supported	OmniSwitch 6250 Series
Functionality Supported	High-level event logging mechanism that forwards requests from applications to enabled logging devices.
Functionality Not Supported	Not intended for debugging individual hardware applications.
Logging Devices	Flash Memory/Console/IP Address
Application ID Levels Supported	IDLE (255), DIAG (0), IPC-DIAG (1), QDRIVER (2), QDISPATCHER (3), IPC-LINK (4), NI-SUPERVISION (5), INTERFACE (6), 802.1Q (7), VLAN (8), GM (9), BRIDGE (10), STP (11), LINKAGG (12), QOS (13), RSVP (14), IP (15), IPMS (17), AMAP (18), GMAP (19), SLB(25), AAA (20), IPC-MON (21), IP-HELPER (22), PMM (23), MODULE (24), EIPC (26), CHASSIS (64), PORT-MGR (65), CONFIG (66), CLI (67), SNMP (68), WEB (69), MIPGW (70), SESSION (71), TRAP (72), POLICY (73), DRC (74), SYSTEM (75), HEALTH (76), NAN-DRIVER (78), RMON (79), TELENET (80), PSM (81), FTP (82), SNMI (83), DISTRIB (84), EPILOGUE (85), LDAP (86), NOSNMP (87), SSL (88), DBGGW (89), LANPOWER (108)
Severity Levels/Types Supported	2 (Alarm - highest severity), 3 (Error), 4 (Alert), 5 (Warning) 6 (Info - default), 7 (Debug 1), 8 (Debug 2), 9 (Debug 3 - lowest severity)

Switch Logging Defaults

The following table shows switch logging default values.

Global Switch Logging Defaults

Parameter Description	CLI Command	Default Value/Comments
Enabling/Disabling switch logging	swlog	Enabled
Switch logging severity level	swlog appid level	Default severity level is info. The numeric equivalent for info is 6
Enabling/Disabling switch logging Output	swlog output	Flash Memory and Console
Switch logging file size	swlog output flash file-size	128000 bytes

Quick Steps for Configuring Switch Logging

- 1 Enable switch logging by using the following command:

```
-> swlog
```

- 2 Specify the ID of the application to be logged along with the logging severity level.

```
-> swlog appid bridge level warning
```

Here, the application ID specifies bridging and the severity is set to the “warning” level.

- 3 Specify the output device to which the switch logging information will be sent.

```
-> swlog output console
```

In this example, the switch logging information will be sent to the console port.

Note. *Optional.* To verify the switch logging configuration, enter the **show swlog** command. The display is similar to the one shown below:

```
Switch Logging is:
  - INITIALIZED
  - RUNNING

Log Device(s)
-----
flash
console

Only Applications not at the level 'info' (6) are shown
Application ID  Level
-----
BRIDGE(10)      warning (5)
```

For more information about this command, or the “Switch Logging Commands” chapter in the *OmniSwitch 6250 CLI Reference Guide*.

Switch Logging Overview

Switch logging uses a formatted string mechanism to process log requests from switch applications. When a log request is received, switch logging compares the severity level included with the request to the severity level stored for the application ID. If there is a match, a log message is generated using the format specified by the log request and placed in the switch log queue. Switch logging then returns control back to the calling application.

You can specify the path to where the log file will be printed in the switch's flash file system. You can also send the log file to other output devices, such as the console or remote IP address. In this case, the log records generated are copied to all configured output devices.

Switch logging information can be displayed and configured through CLI commands, WebView, and SNMP. The information generated by switch logging can be helpful in resolving configuration or authentication issues, as well as general errors.

Notes. Although switch logging provides complementary functionality to switch debugging facilities, the switch logging commands are not intended for use with low-level hardware and software debugging functions.

The **configuration snapshot** command can be used to capture and save all switch logging configuration settings in a text file that can be viewed, edited, and used as a configuration file. See the “Working with Configuration Files” chapter of the *OmniSwitch 6250 Switch Management Guide*.

Switch Logging Commands Overview

This section describes the switch logging CLI commands, for enabling or disabling switch logging, displaying the current status of the switch logging feature, and displaying stored log information.

Enabling Switch Logging

The **swlog** command initializes and enables switch logging, while **no swlog** disables it.

To enable switch logging, enter the **swlog** command:

```
-> swlog
```

To disable switch logging, enter the **no swlog** command:

```
-> no swlog
```

No confirmation message will appear on the screen for either command.

Setting the Switch Logging Severity Level

The switch logging feature can log all switch error-type events for a particular switch application. You can also assign severity levels to the switch applications that will cause some of the events to be filtered out of your display. The **swlog appid level** command is used to assign the severity levels to the applications.

The syntax for the **swlog appid level** command requires that you identify a switch application and assign it a severity level. The severity level controls the kinds of error-type events that will be recorded by the switch logging function. If an application experiences an event equal to or greater than the severity level assigned to the application, the event will be recorded and forwarded to the configured output devices. You can specify the application either by the application ID CLI keyword or by its numeric equivalent.

The application ID information is shown in the following table. The severity level information is shown in the table beginning on [page 39-8](#).

CLI Keyword	Numeric Equivalent	Application ID
IDLE	255	APPID_IDLE
DIAG	0	APPID_DIAGNOSTICS
IPC-DIAG	1	APPID_IPC_DIAGNOSTICS
QDRIVER	2	APPID_QDRIVER
QDISPATCHER	3	APPID_QDISPATCHER
IPC-LINK	4	APPID_IPC_LINK
NI-SUPERVISION	5	APPID_NI_SUP_AND_PROBER
INTERFACE	6	APPID_ESM_DRIVER
802.1Q	7	APPID_802.1Q
VLAN	8	APPID_VLAN_MGR
GM	9	APPID_GROUPMOBILITY (RESERVED)
BRIDGE	10	APPID_SRCLEANING

CLI Keyword	Numeric Equivalent	Application ID
STP	11	APPID_SPANNINGTREE
LINKAGG	12	APPID_LINKAGGREGATION
QOS	13	APPID_QOS
RSVP	14	APPID_RSVP
IP	15	APPID_IP
IPMS	17	APPID_IPMS
AMAP	18	APPID_XMAP
GMAP	19	APPID_GMAP
AAA	20	APPID_AAA
IPC-MON	21	APPID_IPC_MON
IP-HELPER	22	APPID_BOOTP_RELAY
PMM	23	APPID_MIRRORING_MONITORING
MODULE	24	APPID_L3HRE
SLB	25	APPID_SLB
EIPC	26	APPID_EIPC
CHASSIS	64	APPID_CHASSISUPER
PORT-MGR	65	APPID_PORT_MANAGER
CONFIG	66	APPID_CONFIGMANAGER
CLI	67	APPID_CLI
SNMP	68	APPID_SNMP_AGENT
WEB	69	APPID_WEBMGT
MIPGW	70	APPID_MIPGW
SESSION	71	APPID_SESSION_MANAGER
TRAP	72	APPID_TRAP_MANAGER
POLICY	73	APPID_POLICY_MANAGER
DRC	74	APPID_DRC
SYSTEM	75	APPID_SYSTEM_SERVICES
HEALTH	76	APPID_HEALTHMON
NAN-DRIVER	78	APPID_NAN_DRIVER
RMON	79	APPID_RMON
TELNET	80	APPID_TELNET
PSM	81	APPID_PSM
FTP	82	APPID_FTP
SMNI	83	APPID_SMNI
DISTRIB	84	APPID_DISTRIB

CLI Keyword	Numeric Equivalent	Application ID
EPILOGUE	85	APPID_EPILOGUE
LDAP	86	APPID_LDAP
NOSNMP	87	APPID_NOSNMP
SSL	88	APPID_SSL
DBGGW	89	APPID_DBGGW
LANPOWER	108	APPID_LANPOWER

The **level** keyword assigns the error-type severity level to the specified application IDs. Values range from 2 (highest severity) to 9 (lowest severity). The values are defined in the following table:

Severity Level	Type	Description
2 (<i>highest severity</i>)	Alarm	A serious, non-recoverable error has occurred and the system should be rebooted.
3	Error	System functionality is reduced.
4	Alert	A violation has occurred.
5	Warning	An unexpected, non-critical event has occurred.
6 (<i>default</i>)	Info	Any other non-debug message.
7	Debug 1	A normal event debug message.
8	Debug 2	A debug-specific message.
9 (<i>lowest severity</i>)	Debug 3	A maximum verbosity debug message.

Specifying the Severity Level

To specify the switch logging severity level, use the **swlog appid level** command. The application ID can be expressed by using either the ID number or the application ID CLI keyword as listed in the table beginning on [page 39-6](#). The severity level can be expressed by using either the severity level number or the severity level type as shown in the table above. The following syntax assigns the “warning” severity level (or 5) to the “system” application, (ID number 75) by using the severity level and application names.

```
-> swlog appid system level warning
```

The following command makes the same assignment by using the severity level and application numbers.

```
-> swlog appid 75 level 3
```

No confirmation message appears on the screen for either command.

Removing the Severity Level

To remove the switch logging severity level, enter the **no swlog appid level** command, including the application ID and severity level values. The following is a typical example:

```
-> no swlog appid 75 level 5
```

Or, alternatively, as:

```
-> no swlog appid system level warning
```

No confirmation message will appear on the screen.

Specifying the Switch Logging Output Device

The **swlog output** command allows you to send the switch logging information to your console, to the switch's flash memory, or to a specified IP or IPv6 address(es).

Enabling/Disabling Switch Logging Output to the Console

To enable the switch logging output to the console, enter the following command:

```
-> swlog output console
```

To disable the switch logging output to the console, enter the following command:

```
-> no swlog output console
```

No confirmation message will appear on the console screen for either command.

Enabling/Disabling Switch Logging Output to Flash Memory

To enable the switch logging output to flash memory, enter the following:

```
-> swlog output flash
```

To disable the switch logging output to flash memory, enter the following command:

```
-> no swlog output flash
```

No confirmation message will appear on the screen for either command.

Specifying an IP Address for Switch Logging Output

To specify a particular IP address destination (e.g., a server) for switch logging output, enter the **swlog output socket ipaddr** command, specifying the target IP address to which output will be sent. For example, if the target IP address is 168.23.9.100, you would enter:

```
-> swlog output socket ipaddr 168.23.9.100
```

No confirmation message will appear on the screen.

Note. You can also send syslog files to multiple hosts (maximum of four).

Disabling an IP Address from Receiving Switch Logging Output

To disable all configured output IP addresses from receiving switch logging output, enter the following command:

```
-> no swlog output socket
```

No confirmation message will appear on the screen.

To disable a specific configured output IP address from receiving switch logging output, use the same command as above but specify an IPv4 or IPv6 address. For example:

```
-> no swlog output socket 174.16.5.1
```

Displaying Switch Logging Status

You can display the current status of switch logging on your console screen by using the [show swlog](#) command. The following information is displayed:

- The enable/disable status of switch logging.
- A list of current output devices configured for switch logging.
- The switch logging severity level for each application that is not set to the “info” (6) setting.

The following is a sample display:

```
-> show swlog

Switch Logging is:
    - INITIALIZED
    - RUNNING

Log Device(s)
-----
flash
console

Only Applications not at the level 'info' (6) are shown
Application ID   Level
-----
CHASSIS (64)    debug3 (9)

->
```

For this example, switch logging is enabled. Switch logging information is being sent to the switch’s flash memory and to the console. Additionally, the severity level for the chassis application ID has been set to the “debug3” (or “9”) severity level.

Configuring the Switch Logging File Size

By default, the size of the switch logging file is 128000 bytes. To configure the size of the switch logging file, use the **swlog output flash file-size** command. To use this command, enter **swlog output flash file size** followed by the number of bytes, which must be at least 32000. (The maximum size the file can be is dependent on the amount of free memory available in flash memory.)

Note. Use the **ls** command, which is described in the *OmniSwitch 6250 Switch Management Guide*, to determine the amount of available flash memory.

For example, to set the switch logging file to 500000 bytes enter:

```
-> swlog output flash file-size 500000
```

Clearing the Switch Logging Files

You can clear the data stored in the switch logging files by executing the following command:

```
-> swlog clear
```

This command will cause the switch to clear all the switch logging information and begin recording again. As a result, the switch will display a shorter file when you execute the **show log swlog** command. You may want to use **swlog clear** when the switch logging display is too long due to some of the data being old or out of date.

No confirmation message will appear on the screen.

Displaying Switch Logging Records

The **show log swlog** command can produce a display showing *all* the switch logging information or you can display information according to session, timestamp, application ID, or severity level. For details, refer to the *OmniSwitch 6250 CLI Reference Guide*. The following sample screen output shows a display of all the switch logging information.

Note. Switch logging frequently records a very large volume of data. It can take several minutes for all the switch logging information to scroll to the console screen.

```
-> show log swlog
Displaying file contents for 'swlog2.log'
FILEID: fileName[swlog2.log], endPtr[32]
        configSize[64000], currentSize[64000], mode[2]
Displaying file contents for 'swlog1.log'
FILEID: fileName[swlog1.log], endPtr[395]
        configSize[64000], currentSize[64000], mode[1]

Time Stamp                Application      Level   Log Message
-----+-----+-----+-----
MON NOV 11 12:42:11 2005      SYSTEM      info Switch Logging files cleared by
command
MON NOV 11 13:07:26 2005              WEB         info The HTTP session login successfu
l!
MON NOV 11 13:18:24 2005              WEB         info The HTTP session login successfu
l!
MON NOV 11 13:24:03 2005      TELNET      info New telnet connection, Address,
128.251.30.88
MON NOV 11 13:24:03 2005      TELNET      info Session 4, Created
MON NOV 11 13:59:04 2005              WEB         info The HTTP session user logout suc
cessful!
```

The fields in the above example are defined as follows:

- The **FILE ID** field specifies the File name (e.g., swlog1.log), endPtr Global Sequence ID reference number (e.g., 9968), Configuration Size (e.g., 10000), Current Size (e.g., 10000), and Mode (e.g., 2).
- The **Timestamp** field indicates when the swlog entry occurred (e.g., MON, NOV 11, 12:42:11 2005).
- The **Application** field specifies the application ID for which the stored swlog information is displayed (e.g., SYSTEM).
- The **Level** field specifies the severity level for which the stored information is displayed (e.g., Warning).
- The **Log Message** field specifies the condition recorded by the switch logging feature. The information in this field usually wraps around to the next line of the screen display as shown in this example.

A Software License and Copyright Statements

This appendix contains Alcatel-Lucent and third-party software vendor license and copyright statements.

Alcatel-Lucent License Agreement

ALCATEL-LUCENT SOFTWARE LICENSE AGREEMENT

IMPORTANT. Please read the terms and conditions of this license agreement carefully before opening this package.

By opening this package, you accept and agree to the terms of this license agreement. If you are not willing to be bound by the terms of this license agreement, do not open this package. Please promptly return the product and any materials in unopened form to the place where you obtained it for a full refund.

1. **License Grant.** This is a license, not a sales agreement, between you (the “Licensee”) and Alcatel-Lucent. Alcatel-Lucent hereby grants to Licensee, and Licensee accepts, a non-exclusive license to use program media and computer software contained therein (the “Licensed Files”) and the accompanying user documentation (collectively the “Licensed Materials”), only as authorized in this License Agreement. Licensee, subject to the terms of this License Agreement, may use one copy of the Licensed Files on the Licensee’s system. Licensee agrees not to assign, sublicense, transfer, pledge, lease, rent, or share their rights under this License Agreement. Licensee may retain the program media for backup purposes with retention of the copyright and other proprietary notices. Except as authorized under this paragraph, no copies of the Licensed Materials or any portions thereof may be made by Licensee and Licensee shall not modify, decompile, disassemble, reverse engineer, or otherwise attempt to derive the Source Code. Licensee is also advised that Alcatel-Lucent products contain embedded software known as firmware which resides in silicon. Licensee may not copy the firmware or transfer the firmware to another medium.

2. **Alcatel-Lucent’s Rights.** Licensee acknowledges and agrees that the Licensed Materials are the sole property of Alcatel-Lucent and its licensors (herein “its licensors”), protected by U.S. copyright law, trademark law, and are licensed on a right to use basis. Licensee further acknowledges and agrees that all rights, title, and interest in and to the Licensed Materials are and shall remain with Alcatel-Lucent and its licensors and that no such right, license, or interest shall be asserted with respect to such copyrights and trademarks. This License Agreement does not convey to Licensee an interest in or to the Licensed Materials, but only a limited right to use revocable in accordance with the terms of this License Agreement.

3. **Confidentiality.** Alcatel-Lucent considers the Licensed Files to contain valuable trade secrets of Alcatel-Lucent, the unauthorized disclosure of which could cause irreparable harm to Alcatel-Lucent. Except as expressly set forth herein, Licensee agrees to use reasonable efforts not to disclose the Licensed Files to any third party and not to use the Licensed Files other than for the purpose authorized by this License Agreement. This confidentiality obligation shall continue after any termination of this License Agreement.

4. **Indemnity.** Licensee agrees to indemnify, defend and hold Alcatel-Lucent harmless from any claim, lawsuit, legal proceeding, settlement or judgment (including without limitation Alcatel-Lucent's reasonable United States and local attorneys' and expert witnesses' fees and costs) arising out of or in connection with the unauthorized copying, marketing, performance or distribution of the Licensed Files.

5. **Limited Warranty.** Alcatel-Lucent warrants, for Licensee's benefit alone, that the program media shall, for a period of ninety (90) days from the date of commencement of this License Agreement (referred to as the Warranty Period), be free from defects in material and workmanship. Alcatel-Lucent further warrants, for Licensee benefit alone, that during the Warranty Period the Licensed Files shall operate substantially in accordance with the functional specifications in the User Guide. If during the Warranty Period, a defect in the Licensed Files appears, Licensee may return the Licensed Files to Alcatel-Lucent for either replacement or, if so elected by Alcatel-Lucent, refund of amounts paid by Licensee under this License Agreement. EXCEPT FOR THE WARRANTIES SET FORTH ABOVE, THE LICENSED MATERIALS ARE LICENSED "AS IS" AND ALCATEL-LUCENT AND ITS LICENSORS DISCLAIM ANY AND ALL OTHER WARRANTIES, WHETHER EXPRESS OR IMPLIED, INCLUDING (WITHOUT LIMITATION) ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SOME STATES DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES SO THE ABOVE EXCLUSIONS MAY NOT APPLY TO LICENSEE. THIS WARRANTY GIVES THE LICENSEE SPECIFIC LEGAL RIGHTS. LICENSEE MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE.

6. **Limitation of Liability.** Alcatel-Lucent's cumulative liability to Licensee or any other party for any loss or damages resulting from any claims, demands, or actions arising out of or relating to this License Agreement shall not exceed the license fee paid to Alcatel-Lucent for the Licensed Materials. IN NO EVENT SHALL ALCATEL-LUCENT BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL, OR EXEMPLARY DAMAGES OR LOST PROFITS, EVEN IF ALCATEL-LUCENT HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME STATES DO NOT ALLOW THE LIMITATION OR EXCLUSION OF LIABILITY FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION TO INCIDENTAL OR CONSEQUENTIAL DAMAGES MAY NOT APPLY TO LICENSEE.

7. **Export Control.** This product is subject to the jurisdiction of the United States. Licensee may not export or reexport the Licensed Files, without complying with all United States export laws and regulations, including but not limited to (i) obtaining prior authorization from the U.S. Department of Commerce if a validated export license is required, and (ii) obtaining "written assurances" from licensees, if required.

8. **Support and Maintenance.** Except as may be provided in a separate agreement between Alcatel-Lucent and Licensee, if any, Alcatel-Lucent is under no obligation to maintain or support the copies of the Licensed Files made and distributed hereunder and Alcatel-Lucent has no obligation to furnish Licensee with any further assistance, documentation or information of any nature or kind.

9. **Term.** This License Agreement is effective upon Licensee opening this package and shall continue until terminated. Licensee may terminate this License Agreement at any time by returning the Licensed Materials and all copies thereof and extracts therefrom to Alcatel-Lucent and certifying to Alcatel-Lucent in writing that all Licensed Materials and all copies thereof and extracts therefrom have been returned or erased by the memory of Licensee's computer or made non-readable. Alcatel-Lucent may terminate this License Agreement upon the breach by Licensee of any term hereof. Upon such termination by Alcatel-

Lucent, Licensee agrees to return to Alcatel-Lucent or destroy the Licensed Materials and all copies and portions thereof.

10. Governing Law. This License Agreement shall be construed and governed in accordance with the laws of the State of California.

11. Severability. Should any term of this License Agreement be declared void or unenforceable by any court of competent jurisdiction, such declaration shall have no effect on the remaining terms herein.

12. No Waiver. The failure of either party to enforce any rights granted hereunder or to take action against the other party in the event of any breach hereunder shall not be deemed a waiver by that party as to subsequent enforcement of rights or subsequent actions in the event of future breaches.

13. Notes to United States Government Users. Software and documentation are provided with restricted rights. Use, duplication or disclosure by the government is subject to (i) restrictions set forth in GSA ADP Schedule Contract with Alcatel-Lucent's reseller(s), or (ii) restrictions set forth in subparagraph (c) (1) and (2) of 48 CFR 52.227-19, as applicable.

14. Third Party Materials. Licensee is notified that the Licensed Files contain third party software and materials licensed to Alcatel-Lucent by certain third party licensors. Some third party licensors (e.g., Wind River and their licensors with respect to the Run-Time Module) are third party beneficiaries to this License Agreement with full rights of enforcement. Please refer to the section entitled "[Third Party Licenses and Notices](#)" on page A-4 for the third party license and notice terms.

Third Party Licenses and Notices

The licenses and notices related only to such third party software are set forth below:

A. Booting and Debugging Non-Proprietary Software

A small, separate software portion aggregated with the core software in this product and primarily used for initial booting and debugging constitutes non-proprietary software, some of which may be obtained in source code format from Alcatel-Lucent for a limited period of time. Alcatel-Lucent will provide a machine-readable copy of the applicable non-proprietary software to any requester for a cost of copying, shipping and handling. This offer will expire 3 years from the date of the first shipment of this product.

B. The OpenLDAP Public License: Version 2.8, 17 August 2003

Redistribution and use of this software and associated documentation (“Software”), with or without modification, are permitted provided that the following conditions are met:

- 1 Redistributions of source code must retain copyright statements and notices.
- 2 Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
- 3 Redistributions must contain a verbatim copy of this document.
- 4 The names and trademarks of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission.
- 5 Due credit should be given to the OpenLDAP Project.
- 6 The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use the Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND CONTRIBUTORS “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

OpenLDAP is a trademark of the OpenLDAP Foundation.

Copyright 1999-2000 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distributed verbatim copies of this document is granted.

C. Linux

Linux is written and distributed under the GNU General Public License which means that its source code is freely-distributed and available to the general public.

D. GNU GENERAL PUBLIC LICENSE: Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0 This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either

verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term “modification”.) Each licensee is addressed as “you”.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1 You may copy and distribute verbatim copies of the Program’s source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2 You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a** You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b** You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c** If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it. Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3 You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a** Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4 You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5 You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6 Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7 If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on

consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8 If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9 The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10 If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11 BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12 IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS.

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.> Copyright (C)
19yy <name of author>
```

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) 19yy name of author Gnomovision comes with
ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software,
and you are welcome to redistribute it under certain conditions; type 'show c' for details.
```

The hypothetical commands ‘show w’ and ‘show c’ should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than ‘show w’ and ‘show c’; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a “copyright disclaimer” for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision'
(which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

URLWatch:

For notice when this page changes, fill in your email address.

Maintained by: Webmaster, Linux Online Inc.

Last modified: 09-Aug-2000 02:03AM.

Views since 16-Aug-2000: 177203.

Material copyright Linux Online Inc.
Design and compilation copyright (c)1994-2002 Linux Online Inc.
Linux is a registered trademark of Linus Torvalds
Tux the Penguin, featured in our logo, was created by Larry Ewing
Consult our privacy statement

URLWatch provided by URLWatch Services.
All rights reserved.

E. University of California

Provided with this product is certain TCP input and Telnet client software developed by the University of California, Berkeley.

Copyright (C) 1987. The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

F. Carnegie-Mellon University

Provided with this product is certain BOOTP Relay software developed by Carnegie-Mellon University.

G. Random.c

PR 30872 B Kesner created May 5 2000

PR 30872 B Kesner June 16 2000 moved batch_entropy_process to own task iWhirlpool to make code more efficient

random.c -- A strong random number generator

Version 1.89, last modified 19-Sep-99

Copyright Theodore Ts'o, 1994, 1995, 1996, 1997, 1998, 1999. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, and the entire permission notice in its entirety, including the disclaimer of warranties.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission. ALTERNATIVELY, this product may be distributed under the terms of the GNU Public License, in which case the provisions of the GPL are required INSTEAD OF the

above restrictions. (This clause is necessary due to a potential bad interaction between the GPL and the restrictions contained in a BSD-style copyright.)

THIS SOFTWARE IS PROVIDED “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ALL OF WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF NOT ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

H. Apptitude, Inc.

Provided with this product is certain network monitoring software (“MeterWorks/RMON”) licensed from Apptitude, Inc., whose copyright notice is as follows: Copyright (C) 1997-1999 by Apptitude, Inc. All Rights Reserved. Licensee is notified that Apptitude, Inc. (formerly, Technically Elite, Inc.), a California corporation with principal offices at 6330 San Ignacio Avenue, San Jose, California, is a third party beneficiary to the Software License Agreement. The provisions of the Software License Agreement as applied to MeterWorks/RMON are made expressly for the benefit of Apptitude, Inc., and are enforceable by Apptitude, Inc. in addition to Alcatel-Lucent. IN NO EVENT SHALL APPTITUDE, INC. OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES, INCLUDING COSTS OF PROCUREMENT OF SUBSTITUTE PRODUCTS OR SERVICES, LOST PROFITS, OR ANY SPECIAL, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, ARISING IN ANY WAY OUT OF THIS AGREEMENT.

I. Agranat

Provided with this product is certain web server software (“EMWEB PRODUCT”) licensed from Agranat Systems, Inc. (“Agranat”). Agranat has granted to Alcatel-Lucent certain warranties of performance, which warranties [or portion thereof] Alcatel-Lucent now extends to Licensee. IN NO EVENT, HOWEVER, SHALL AGRANAT BE LIABLE TO LICENSEE FOR ANY INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES OF LICENSEE OR A THIRD PARTY AGAINST LICENSEE ARISING OUT OF, OR IN CONNECTION WITH, THIS DISTRIBUTION OF EMWEB PRODUCT TO LICENSEE. In case of any termination of the Software License Agreement between Alcatel-Lucent and Licensee, Licensee shall immediately return the EMWEB Product and any back-up copy to Alcatel-Lucent, and will certify to Alcatel-Lucent in writing that all EMWEB Product components and any copies of the software have been returned or erased by the memory of Licensee’s computer or made non-readable.

J. RSA Security Inc.

Provided with this product is certain security software (“RSA Software”) licensed from RSA Security Inc. RSA SECURITY INC. PROVIDES RSA SOFTWARE “AS IS” WITHOUT ANY WARRANTY WHATSOEVER. RSA SECURITY INC. DISCLAIMS ALL WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO ANY MATTER WHATSOEVER INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS.

K. Sun Microsystems, Inc.

This product contains Coronado ASIC, which includes a component derived from designs licensed from Sun Microsystems, Inc.

L. Wind River Systems, Inc.

Provided with this product is certain software (“Run-Time Module”) licensed from Wind River Systems, Inc. Licensee is prohibited from: (i) copying the Run-Time Module, except for archive purposes consistent with Licensee’s archive procedures; (ii) transferring the Run-Time Module to a third party apart from the product; (iii) modifying, decompiling, disassembling, reverse engineering or otherwise attempting to derive the source code of the Run-Time Module; (iv) exporting the Run-Time Module or underlying technology in contravention of applicable U.S. and foreign export laws and regulations; and (v) using the Run-Time Module other than in connection with operation of the product. In addition, please be advised that: (i) the Run-Time Module is licensed, not sold and that Alcatel-Lucent and its licensors retain ownership of all copies of the Run-Time Module; (ii) WIND RIVER DISCLAIMS ALL IMPLIED WARRANTIES, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, (iii) The SOFTWARE LICENSE AGREEMENT EXCLUDES LIABILITY FOR ANY SPECIAL, INDIRECT, PUNITIVE, INCIDENTAL AND CONSEQUENTIAL DAMAGES; and (iv) any further distribution of the Run-Time Module shall be subject to the same restrictions set forth herein. With respect to the Run-Time Module, Wind River and its licensors are third party beneficiaries of the License Agreement and the provisions related to the Run-Time Module are made expressly for the benefit of, and are enforceable by, Wind River and its licensors.

M. Network Time Protocol Version 4

The following copyright notice applies to all files collectively called the Network Time Protocol Version 4 Distribution. Unless specifically declared otherwise in an individual file, this notice applies as if the text was explicitly included in the file.

```

*****
*
* Copyright (c) David L. Mills 1992-2003
*
* Permission to use, copy, modify, and distribute this software and
* its documentation for any purpose and without fee is hereby
* granted, provided that the above copyright notice appears in all
* copies and that both the copyright notice and this permission
* notice appear in supporting documentation, and that the name
* University of Delaware not be used in advertising or publicity
* pertaining to distribution of the software without specific,
* written prior permission. The University of Delaware makes no
* representations about the suitability this software for any
* purpose. It is provided "as is" without express or implied
* warranty.
*
*****

```

N.Remote-ni

Provided with this product is a file (part of GDB), the GNU debugger and is licensed from Free Software Foundation, Inc., whose copyright notice is as follows: Copyright (C) 1989, 1991, 1992 by Free Software Foundation, Inc. Licensee can redistribute this software and modify it under the terms of General Public License as published by Free Software Foundation Inc.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

O.GNU Zip

GNU Zip -- A compression utility which compresses the files with zip algorithm.

Copyright (C) 1992-1993 Jean-loup Gailly.

BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

P. FREESCALE SEMICONDUCTOR SOFTWARE LICENSE AGREEMENT

Provided with this product is a software also known as DINK32 (Dynamic Interactive Nano Kernel for 32-bit processors) solely in conjunction with the development and marketing of your products which use and incorporate microprocessors which implement the PowerPC (TM) architecture manufactured by Motorola. The licensee comply with all of the following restrictions:

1. This entire notice is retained without alteration in any modified and/or redistributed versions.
2. The modified versions are clearly identified as such. No licenses are granted by implication, estoppel or otherwise under any patents or trademarks of Motorola, Inc.

The SOFTWARE is provided on an "AS IS" basis and without warranty. To the maximum extent permitted by applicable law, MOTOROLA DISCLAIMS ALL WARRANTIES WHETHER EXPRESS OR IMPLIED, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY AGAINST INFRINGEMENT WITH REGARD TO THE SOFTWARE (INCLUDING ANY MODIFIED VERSIONS THEREOF) AND ANY ACCOMPANYING WRITTEN MATERIALS. To the maximum extent permitted by applicable law, IN NO EVENT SHALL MOTOROLA BE LIABLE FOR ANY DAMAGES WHATSOEVER.

Copyright (C) Motorola, Inc. 1989-2001 All rights reserved.

Version 13.1

Q. Boost C++ Libraries

Provided with this product is free peer-reviewed portable C++ source libraries.

Version 1.33.1

Copyright (C) by Beman Dawes, David Abrahams, 1998-2003. All rights reserved.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,

ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

R. U-Boot

Provided with this product is a software licensed from Free Software Foundation Inc. This is used as OS Bootloader; and located in on-board flash. This product is standalone and not linked (statically or dynamically) to any other software.

Version 1.1.0

Copyright (C) 2000-2004. All rights reserved.

S. Solaris

Provided with this product is free software; Licensee can redistribute it and/or modify it under the terms of the GNU General Public License.

Copyright (C) 1992-1993 Jean-loup Gailly. All rights reserved.

T. Internet Protocol Version 6

Copyright (C) 1982, 1986, 1990, 1991, 1993. The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the University of California, Berkeley and its contributors.

4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION). HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The copyright of the products such as crypto, dhcp, net, netinet, netinet6, netley, netwrs, libinet6 are same as that of the internet protocol version 6.

U. CURSES

Copyright (C) 1987. The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission.

V. ZModem

Provided with this product is a program or code that can be used without any restriction.

Copyright (C) 1986 Gary S. Brown. All rights reserved.

W.Boost Software License

Provided with this product is reference implementation, so that the Boost libraries are suitable for eventual standardization. Boost works on any modern operating system, including UNIX and Windows variants.

Version 1.0

Copyright (C) Gennadiy Rozental 2005. All rights reserved.

X. OpenLDAP

Provided with this software is an open source implementation of the Lightweight Directory Access Protocol (LDAP).

Version 3

Copyright (C) 1990, 1998, 1999, Regents of the University of Michigan, A. Hartgers, Juan C. Gomez. All rights reserved.

This software is not subject to any license of Eindhoven University of Technology. Redistribution and use in source and binary forms are permitted only as authorized by the OpenLDAP Public License.

This software is not subject to any license of Silicon Graphics Inc. or Purdue University. Redistribution and use in source and binary forms are permitted without restriction or fee of any kind as long as this notice is preserved.

Y. BITMAP.C

Provided with this product is a program for personal and non-profit use.

Copyright (C) Allen I. Holub, All rights reserved.

Z. University of Toronto

Provided with this product is a code that is modified specifically for use with the STEVIE editor. Permission is granted to anyone to use this software for any purpose on any computer system, and to redistribute it freely, subject to the following restrictions:

1. The author is not responsible for the consequences of use of this software, no matter how awful, even if they arise from defects in it.
2. The origin of this software must not be misrepresented, either by explicit claim or by omission.
3. Altered versions must be plainly marked as such, and must not be misrepresented as being the original software.

Version 1.5

Copyright (C) 1986 by University of Toronto and written by Henry Spencer.

AA.Free/OpenBSD

Copyright (c) 1982, 1986, 1990, 1991, 1993 The Regents of University of California. All Rights Reserved.

Index

qos log lines command 34-19
qos port servicing mode command 34-26
qos stats interval command 34-22

Numerics

10/100/1000 ports
 defaults 1-3
802.1AB 20-1
 defaults 20-2
 specifications 20-2
 verify information about 20-19
802.1p
 trusted ports 34-28
802.1Q 22-1
 application examples 22-8
 defaults 22-2
 enabling notification 20-15
 enabling tagging 22-5
 frame type 22-6
 overview 22-3
 specifications 22-2
 trusted ports 34-5, 34-29
 verify information about 22-10
802.1Q ports
 trusted 34-28
802.1X 30-7, 32-1
 accounting 32-8
 and DHCP 32-7
 components 32-6
 defaults 32-2
 port authorization 32-10
 port parameters 30-13, 32-10
 port timeouts 32-10
 re-authentication 32-7, 32-11
 specifications 30-2, 32-2
802.1x command 32-2
802.1x initialize command 32-12
802.1x re-authenticate command 32-12
802.3ad
 see dynamic link aggregation

A

aaa accounting 802.1x command 32-12
aaa ace-server clear command 31-7
aaa authentication 802.1x command 32-9
 and 802.1X port behavior 32-7
aaa ldap-server command
 LDAP authentication 31-26
aaa radius-server command 32-9

 RADIUS authentication 31-12, 31-15
Access Control Lists
 see ACLs
access list 27-15
 creating 27-15
ACE/Server
 for authentication 31-7
ACLs
 application examples 35-4, 35-21
 bridged traffic 35-6
 defaults 35-3
 disposition 35-5, 35-7
 Layer 2 35-10
 Layer 2 application examples 35-11
 Layer 3 35-11
 Layer 3 application examples 35-12
 multicast 35-13
 security features 35-15
 verify information about 35-19
actions
 combined with conditions 34-9, 34-11
 creating policy actions 34-34
 for ACLs 35-10
Address Resolution Protocol
 see ARP
advertisements 28-6
 destination address 28-9
 IP address preference 28-10
 lifetime 28-10
 transmission interval 28-9
Alcatel Mapping Adjacency Protocol 21-1
alerts 39-8
AMAP
 see Alcatel Mapping Adjacency Protocol
amap common time command 21-6
amap disable command 21-5
amap discovery time command 21-5
amap enable command 21-5
Application example
 Learned Port Security Configuration 3-3
application example
 Ethernet OAM 16-3
 MST 10-14
 MSTI 10-16
application examples
 802.1Q 22-8
 ACLs 35-4
 assigning ports to VLANs 6-3
 authentication servers 31-4
 Configuring 802.1AB 20-4
 DHCP Relay 29-4, 29-6, 29-8
 dynamic link aggregation 24-4, 24-29
 GVRP 5-5
 ICMP policies 34-68
 interswitch protocols 21-8
 IP 25-4
 IPMS 36-34, 36-36
 IPv6 26-4
 Layer 2 ACLs 35-11

- Layer 3 ACLs 35-12
 - mobile ports 6-3, 6-6, 6-8
 - policies 34-62
 - policy map groups 34-56
 - Port Mapping 7-2, 7-6
 - port mirroring 38-4
 - port monitoring 38-6, 38-8
 - QoS 34-31, 34-62
 - RDP 28-3
 - RIP 27-3
 - RMON 38-11
 - source learning 2-3
 - Spanning Tree Algorithm and Protocol 11-10, 11-40
 - static link aggregation 23-3, 23-11
 - switch health 38-13
 - switch logging 39-4
 - UDLD 18-3
 - VLAN advertisements 5-4
 - VLAN rules 8-3, 8-14
 - VLANs 4-3, 4-12, 6-3
 - applied configuration 34-59
 - how to verify 34-61
 - ARP
 - clearing the ARP cache 25-13
 - creating a permanent entry 25-12
 - deleting a permanent entry 25-12
 - dynamic entry 25-12
 - filtering 25-14
 - local proxy 25-13
 - arp** command 25-12
 - arp filter** command 25-14
 - assigning ports 4-7
 - assigning ports to VLANs 6-1
 - application examples 6-3
 - defaults 6-2
 - dynamic port assignment 6-4
 - static port assignment 6-4
 - Authenticated Switch Access
 - LDAP VSAs 31-22
 - authentication servers
 - application example 31-4
 - defaults 31-3
 - how backups work 31-5
 - see* LDAP authentication servers, RADIUS authentication servers
 - automatic IP configuration 29-12
- B**
- boundary port 10-12
 - BPDU
 - see* Bridge Protocol Data Units
 - bridge 1x1 forward delay** command 11-23
 - bridge 1x1 hello time** command 11-22
 - bridge 1x1 protocol** command 11-20
 - bridge 1x1 slot/port** command 11-29
 - bridge 1x1 slot/port admin-edge** command 11-36
 - bridge 1x1 slot/port path cost** command 11-32
 - bridge auto-vlan-containment** command 11-25
 - bridge cist forward delay** command 11-23
 - bridge cist hello time** command 11-22
 - bridge cist protocol** command 11-20
 - bridge cist slot/port admin-edge** command 11-36
 - bridge forward delay** command 11-23
 - bridge hello time** command 11-22
 - bridge max age** command 11-22
 - bridge mode** command 11-12
 - bridge msti priority** command 11-21
 - bridge path cost mode** command 11-24
 - bridge priority** command 11-21
 - bridge protocol** command 11-20
 - Bridge Protocol Data Units
 - contents 11-8
 - bridge slot/port** command 11-24
 - bridge slot/port connection** command 11-35
 - bridge slot/port path cost** command 11-32
 - bridge slot/port priority** command 11-30
 - built-in port groups 34-14
 - used with Policy Based Routing 34-70
- C**
- clear arp filter** command 25-14
 - clear arp-cache** command 25-13
 - combo ports 1-4
 - configuring 1-17
 - defaults 1-3
 - overview 1-4
 - condition groups
 - for ACLs 34-46, 35-8
 - MAC groups 34-50, 34-53
 - network groups 34-47
 - port groups 34-51
 - sample configuration 34-46
 - service groups 34-49
 - verify information about 34-55
 - conditions
 - combined with actions 34-9, 34-11
 - configuring 34-33
 - for ACLs 35-9
 - how to create 34-33
 - see also* condition groups
 - testing before applying 34-43
 - valid combinations 34-7
 - valid combinations for ACLs 35-6
 - Configuring 802.1AB
 - application examples 20-4
- D**
- debug messages 39-8
 - debug qos** command 34-18
 - default route
 - IP 25-11
 - defaults
 - 10/100/1000 ports 1-3
 - 802.1AB 20-2
 - 802.1Q 22-2
 - 802.1X 32-2

- ACLs 35-3
 - assigning ports to VLANs 6-2
 - authentication servers 31-3
 - combo ports 1-3
 - DHCP Relay 29-3
 - DVMRP 5-2
 - dynamic link aggregation 24-3
 - Ethernet OAM 15-2, 16-2
 - Ethernet ports 1-2, 1-3
 - interswitch protocols 21-2
 - IP 25-3
 - IPMS 36-4, 36-5
 - IPv6 26-3
 - Learned Port Security 3-2
 - mobile ports 6-2
 - Multiple Spanning Tree 11-5
 - policy servers 33-2
 - Port Mapping 7-2
 - port mirroring 38-3
 - port monitoring 38-5, 38-7
 - QoS 34-12
 - RDP 28-2
 - RDP interface 28-8
 - RIP 27-2
 - RMON 38-11
 - RRSTP 11-5
 - source learning 2-2
 - Spanning Tree Bridge 11-4, 12-2
 - Spanning Tree Port 11-4
 - static link aggregation 23-2
 - switch health 38-13
 - switch logging 39-3
 - UDLD 18-2
 - VLAN rules 8-2
 - VLANs 4-2
 - Denial of Service
 - see* DoS
 - DHCP 29-6
 - used with 802.1X 32-7
 - DHCP Relay 29-1, 29-10
 - application examples 29-4, 29-6, 29-8
 - AVLAN forwarding option 29-11
 - defaults 29-3
 - DHCP server IP address 29-9
 - forward delay time 29-10
 - maximum number of hops 29-11
 - standard forwarding option 29-11
 - statistics 29-26
 - DHCP VLAN rules 8-5
 - directed broadcast 25-23
 - disposition 35-10
 - ACLs 35-5, 35-7
 - global defaults for QoS rules 34-16
 - DoS 25-23
 - enabling traps 25-27
 - setting decay value 25-27
 - setting penalty values 25-26
 - Setting Port Scan Penalty Value 25-27
 - DSCP
 - trusted ports 34-28
 - DVMRP
 - defaults 5-2
 - dynamic link aggregation 24-1
 - application examples 24-4, 24-29
 - defaults 24-3
 - group actor administrative key 24-15
 - group actor system ID 24-16
 - group actor system priority 24-16
 - group administrative state 24-15
 - group partner administrative key 24-17
 - group partner system ID 24-18
 - group partner system priority 24-17
 - groups 24-11
 - assigning ports 24-12
 - creating groups 24-11
 - deleting groups 24-11
 - group names 24-14
 - removing ports 24-13
 - LACPDU bit settings 24-19, 24-23
 - LACPDU frames 24-19, 24-23
 - Link Aggregation Control Protocol (LACP) 24-7
 - MAC address 24-16, 24-18, 24-20, 24-25
 - port actor administrative priority 24-21
 - port actor port priority 24-22
 - port actor system administrative states 24-19
 - port actor system ID 24-20
 - port partner administrative key 24-25
 - port partner administrative priority 24-27
 - port partner administrative state 24-23
 - port partner administrative system ID 24-25
 - port partner administrative system priority 24-26
 - port partner port administrative status 24-27
 - ports 24-12
 - specifications 24-2
 - verify information about 24-32
 - dynamic log
 - LDAP accounting servers 31-25
 - dynamic VLAN port assignment
 - mobile ports 6-4
 - secondary VLANs 6-13
 - VLAN rules 8-1
- ## E
- errors 39-8
 - Ethernet
 - defaults 1-2, 1-3
 - flood rate 1-9
 - frame size 1-11
 - full duplex 1-12, 1-18
 - half duplex 1-12, 1-18
 - multicast traffic 1-9
 - specifications 1-2
 - verify information 1-22
 - Ethernet OAM
 - application example 16-3

- configuration 15-9, 16-4, 38-48
 - Connectivity Fault Management
 - Continuity Check Messages 15-5
 - Link Trace Messages 15-5
 - Loop-back Messages 15-5
 - defaults 15-2, 16-2
 - overview 15-3
 - specifications 15-2, 16-2
 - verification 15-14
 - ethoam association ccm-interval** command 15-10
 - ethoam association** command 15-8
 - ethoam association mhf** command 15-10, 15-11
 - ethoam association-default** command 15-10
 - ethoam domain** command 15-8
 - ethoam end-point** command 15-8
 - ethoam intermediate-point** command 15-8
 - ethoam linktrace** command 15-12
 - ethoam loopback** command 15-11
- F**
- Fast Spanning Tree 11-6
 - filtering lists
 - see* ACLs
 - flow** command 1-15, 1-20
 - frame type 22-6
- G**
- GARP
 - active member 5-3
 - messages 5-3
 - passive member 5-3
 - Generic Attribute Registration Protocol
 - see* GARP
 - GVRP
 - application examples 5-5
 - display configuration on specified port 5-13
 - specifications 5-2
 - gvrp applicant** command 5-10
 - gvrp enable-vlan-advertisement** command 5-12
 - gvrp enable-vlan-registration** command 5-11
 - gvrp maximum vlan** command 5-8
 - gvrp port** command 5-5
 - gvrp registration** command 5-9
 - gvrp static-vlan restrict** command 5-5
 - GVRP Timers 5-10
 - gvrp transparent switching** command 5-8
 - gvrp** command 5-5
- H**
- health interval** command 15-12, 38-45
 - health statistics reset** command 38-47
 - health threshold** command 38-43
 - health threshold limits
 - displaying 38-44
- I**
- ICMP 25-29
 - control 25-32
 - QoS policies for 34-68
 - statistics 25-32
 - icmp messages** command 25-31
 - icmp type** command 25-30, 25-31
 - IEEE 22-1
 - IGMP
 - multicast ACLs 35-1, 35-13
 - IGMP Spoofing 36-18
 - Institute of Electrical and Electronics Engineers
 - see* IEEE
 - interfaces admin** command 1-8
 - interfaces alias** command 1-11
 - interfaces autoneg** command 1-14
 - interfaces crossover** command 1-15
 - interfaces duplex** command 1-12
 - interfaces flood multicast** command 1-9
 - interfaces flood rate** command 1-10
 - interfaces hybrid autoneg** command 1-18
 - interfaces hybrid crossover** command 1-19
 - interfaces hybrid duplex** command 1-18
 - interfaces hybrid speed** command 1-17
 - interfaces ifg** command 1-13
 - interfaces max frame** command 1-11
 - interfaces no l2 statistics** command 1-8
 - interfaces speed** command 1-12
 - inter-frame gap value 1-13
 - Internet Control Message Protocol
 - see* ICMP
 - interswitch protocols
 - AMAP 21-1, 21-3
 - application examples 21-8
 - defaults 21-2
 - specifications 21-2
 - IP 25-1
 - application examples 25-4
 - ARP 25-12
 - defaults 25-3
 - directed broadcast 25-23
 - ICMP 25-29
 - ping 25-32
 - protocols 25-5
 - router ID 25-15
 - router port 25-8
 - router primary address 25-15
 - specifications 25-3
 - static route 25-10, 26-13
 - tracing an IP route 25-33
 - TTL value 25-16
 - UDP 25-33
 - verify information about 25-34
 - ip access-list address** command 27-15
 - ip access-list** command 27-15
 - ip default-ttl** command 25-16
 - ip directed-broadcast** command 25-23
 - ip dos scan close-port-penalty** command 25-26

- ip dos scan decay** command 25-27
 - ip dos scan tcp open-port-penalty** command 25-26
 - ip dos scan threshold** command 25-27
 - ip dos scan udp open-port-penalty** command 25-26
 - ip dos trap** command 25-27
 - ip helper address** command 29-9
 - ip helper boot-up** command 29-12
 - ip helper forward delay** command 29-10
 - ip helper maximum hops** command 29-11
 - ip helper per-vlan** command 29-11
 - ip helper standard** command 29-11
 - ip interface** command 27-3
 - ip load rip** command 27-3, 27-6
 - ip multicast igmp-proxy-version** command 36-9, 36-23
 - ip multicast neighbor-timeout** command 36-9, 36-15, 36-16, 36-17, 36-23, 36-30
 - ip multicast query-interval** command 36-13, 36-14, 36-27
 - ip multicast static-member** command 36-11
 - ip multicast static-neighbor** command 36-24
 - ip multicast static-querier** command 36-11
 - IP Multicast Switching
 - see* IPMS
 - ip multicast switching** command 36-8, 36-18, 36-22, 36-32
 - IP multinetting 25-7
 - ip redist** command 27-12
 - ip rip force-holddowntimer** command 27-9
 - ip rip garbage-timer** command 27-10
 - ip rip holddown-timer** command 27-10
 - ip rip host-route** command 27-11
 - ip rip interface auth-key** command 27-18
 - ip rip interface auth-type** command 27-18
 - ip rip interface** command 27-3, 27-7
 - ip rip interface metric** command 27-8
 - ip rip interface recv-version** command 27-8
 - ip rip interface send-version** command 27-7
 - ip rip interface status** command 27-3, 27-7
 - ip rip invalid-timer** command 27-10
 - ip rip route-tag** command 27-9
 - ip rip status** command 27-3, 27-7
 - ip rip update-interval** command 27-9
 - ip route-pref** command 25-15
 - IP router ports 25-8
 - modifying 25-9
 - removing 25-9
 - ip router primary-address** command 25-15
 - ip router router-id** command 25-15
 - ip router-discovery** command 28-3, 28-8
 - ip router-discovery interface advertisement-address** command 28-9
 - ip router-discovery interface advertisement-lifetime** command 28-10
 - ip router-discovery interface max-advertisement-interval** command 28-9
 - ip router-discovery interface min-advertisement-interval** command 28-10
 - ip router-discovery interface preference-level** command 28-10
 - ip service** command 25-28
 - ip slb admin** command 37-9
 - ip static-route** command 25-10, 26-13
 - IPMS 36-1
 - adding static members 36-12
 - adding static neighbors 36-10
 - adding static queriers 36-11
 - application examples 36-34, 36-36
 - defaults 36-4, 36-5
 - deleting static members 36-12, 36-26
 - deleting static neighbors 36-11
 - deleting static queriers 36-11, 36-25
 - displaying 36-38, 36-39
 - enabling 36-8, 36-18, 36-19, 36-32, 36-33
 - IGMPv2 36-10, 36-24
 - IGMPv3 36-10, 36-23
 - neighbor timeout 36-15, 36-16, 36-17, 36-29, 36-31
 - overview 36-6
 - query interval 36-13, 36-14, 36-27, 36-28
 - RFCs 36-3
 - specifications 36-3
 - IPMV
 - ipv4, ipv6 address 37-15
 - IPv6 26-1
 - addressing 26-6
 - application examples 26-4
 - autoconfiguration of addresses 26-8
 - defaults 26-3
 - specification 26-2
 - verify information about 26-21
 - ipv6 access-list address** command 27-15
 - ipv6 access-list** command 27-15
 - ipv6 address** command 26-4, 26-11
 - ipv6 interface** command 26-4, 26-9
 - ipv6 load rip** command 26-4
 - ipv6 rip interface** command 26-4
 - ipv6 route-pref** command 26-14
- ## J
- jumbo frames 1-2
- ## L
- LACP
 - see* dynamic link aggregation
 - lacp agg actor admin key** command 24-4, 24-12
 - lacp agg actor admin state** command 24-19
 - lacp agg actor port priority** command 24-22
 - lacp agg actor system id** command 24-20
 - lacp agg actor system priority** command 24-21
 - lacp agg partner admin key** command 24-25
 - lacp agg partner admin port** command 24-27
 - lacp agg partner admin port priority** command 24-27
 - lacp agg partner admin state** command 24-23
 - lacp agg partner admin system id** command 24-25
 - lacp agg partner admin system priority** command 24-26
 - lacp linkagg actor admin key** command 24-15
 - lacp linkagg actor system id** command 24-16
 - lacp linkagg actor system priority** command 24-16
 - lacp linkagg admin state** command 24-15
 - lacp linkagg name** command 24-14

- lacp linkagg partner admin key** command 24-17
 - lacp linkagg partner system id** command 24-18
 - lacp linkagg partner system priority** command 24-17
 - lacp linkagg size** command 24-4, 24-11
 - Layer 2
 - statistics counters 1-8
 - LDAP accounting servers
 - dynamic log 31-25
 - standard attributes 31-23
 - LDAP authentication servers
 - directory entries 31-18
 - functional privileges 31-22
 - passwords for 31-21
 - schema extensions 31-18
 - SNMP attributes on authentication servers 31-23
 - SSL 31-27
 - VSAs for Authenticated Switch Access 31-22
 - LDAP servers
 - see* policy servers
 - used for QoS policies 33-3
 - Learned Port Security
 - database table 3-6
 - defaults 3-2
 - disabling 3-7
 - enabling 3-7
 - overview 3-4
 - specifications 3-2
 - Learned Port Security Configuration
 - Application example 3-3
 - Lightweight Directory Access Protocol
 - see* LDAP servers
 - line speed 1-12, 1-17
 - link aggregation
 - 802.1Q 22-5
 - dynamic link aggregation 24-1
 - enabling tagging 22-5
 - Spanning Tree parameters 11-29, 11-31, 11-32, 11-34, 11-36
 - static link aggregation 23-1
 - lldp lldpdu** command 20-4
 - lldp notification** command 20-4
 - lldp tlv dot1** command 20-16
 - lldp tlv dot3** command 20-17
 - lldp tlv management** command 20-4
 - lldp tlv med** command 20-17
 - logged events
 - detail level 34-19
 - sent to PolicyView 34-20
 - types of events 34-18
- ## M
- MAC address table 2-1, 2-5
 - aging time 2-9
 - duplicate MAC addresses 2-5
 - learned MAC addresses 2-5
 - static MAC addresses 2-5
 - MAC address VLAN rules 8-5
 - MAC addresses
 - aging time 2-9, 11-23
 - dynamic link aggregation 24-16, 24-18, 24-20, 24-25
 - learned 2-5
 - statically assigned 2-5
 - mac-address-table** command 2-5
 - mac-address-table-aging-time** command 2-9
 - map groups 34-56
 - application 34-68
 - creating 34-57
 - verifying information 34-58
 - MLD Zapping 36-32
 - mobile port properties 6-16
 - BPDU ignore 6-11
 - default VLAN membership 6-12
 - restore default VLAN 6-12
 - mobile ports 6-11
 - application examples 6-3, 6-6, 6-8
 - defaults 6-2
 - dynamic VLAN port assignment 6-4, 6-12
 - secondary VLANs 6-13
 - trusted 34-5, 34-28
 - VLAN rules 8-1
 - MST 10-4
 - application example 10-14
 - Internal Spanning Tree (IST) Instance 10-9
 - Interoperability 10-12
 - Migration 10-12, 10-13
 - MSTI 10-7
 - application example 10-16
 - MSTP 10-4
 - Multiple Spanning Tree Region 10-8
 - Multicast Listener Discovery (MLD) 36-23
 - Multiple Spanning Tree
 - defaults 11-5
- ## N
- network address VLAN rules 8-5
 - non combo ports
 - configuring 1-12
- ## O
- OSPF redistribution policies
 - deleting 25-19, 25-21, 26-17, 26-19, 27-16
- ## P
- pending configuration 34-59
 - pending policies
 - deleting 34-60
 - testing 34-43
 - Per VLAN DHCP 29-9
 - ping
 - IP 25-32
 - ping** command 25-32
 - policies
 - application examples 34-62
 - applied 34-59
 - built-in 34-14

- conditions 34-33
- creating policy actions 34-34
- how the switch uses them 34-4
- Policy Based Routing 34-70
- precedence 34-37, 35-6
- redirect linkagg 34-67
- redirect port 34-67
- rules 34-35
- verify information about 34-42
- policies configured via PolicyView 34-61
- policy
 - for ACLs 35-10
 - policy actions 35-10
 - policy conditions 35-9
 - policy rule 35-10
- policy action 802.1p** command 34-29
- policy action** command 34-23, 34-31
- policy action map** command 34-56
- policy action redirect linkagg** command 34-67
- policy action redirect port** command 34-67
- policy actions
 - see* actions
- Policy Based Routing 34-70
- policy condition** command 34-31
- policy conditions
 - see* conditions
- policy mac group** command 34-46, 35-8
- policy MAC groups 34-50, 34-53
- policy map group** command 34-56
- policy map groups
 - application example 34-56
- policy network group** command 34-46, 35-8
- policy network groups 34-47
 - switch** default group 34-14, 34-47
- policy port group** command 34-46, 35-8
- policy port groups 34-51
- policy rule** command 34-31
- policy server** command 33-2, 33-4
- policy server flush** command 33-6
 - compared to **qos flush** command 33-7
- policy server load** command 33-6
- policy servers
 - defaults 33-2
 - downloading policies 33-6
 - installing 33-3
 - SSL 33-6
- policy service** command 35-8
- policy service group** command 34-46, 35-8
- policy service groups 34-49
- policy services 34-48
- PolicyView
 - LDAP policy servers 33-1
- Port Based Network Access Control
 - see* 802.1X
- Port Mapping 7-1
 - application examples 7-2, 7-6
 - defaults 7-2
 - specifications 7-2
- port mapping** command 7-2
- Port Mapping Session
 - creating and deleting 7-3
 - enabling and disabling 7-4
- port mirroring 16-2, 38-14
 - application examples 38-4
 - defaults 38-3
 - direction 38-20
 - disabling mirroring status 38-19
 - displaying status 38-21
 - enabling or disabling mirroring status 38-19
 - N-to-1 port mirroring 38-18
 - specifications 38-3
 - unblocking ports 38-19
- port mirroring** command 38-21
- port mirroring session
 - creating 38-18
 - deleting 38-21
 - enabling/disabling 38-20
- port mirroring source** command 38-6
- port mirroring source destination** command 38-18, 38-19, 38-20
- port mobility
 - see* mobile ports
- port monitoring
 - application examples 38-6, 38-8
 - configuring 38-25, 38-30, 38-31
 - creating a data file 38-26
 - defaults 38-5, 38-7
 - deleting a session 38-25, 38-33
 - direction 38-27
 - disabling a session 38-25
 - displaying status and data 38-28, 38-31, 38-33
 - enabling a session 38-25
 - file overwriting 38-27
 - file size 38-26
 - overview 38-24, 38-29
 - pausing a session 38-26
 - resuming a session 38-26
 - session persistence 38-26
 - specifications 38-5, 38-7
 - suppressing file creation 38-27
- port monitoring** command 38-25, 38-26
- port monitoring source** command 38-25, 38-26, 38-27, 38-30
- port VLAN rules 8-6
- ports
 - 802.1Q 22-5
 - displaying QoS information about 34-30
 - enabling tagging 22-5
 - mobile ports 6-11
 - Spanning Tree parameters 11-26
 - trusted 34-28
 - VLAN assignment 4-7, 6-1
- port-security** command 3-7
- port-security shutdown** command 3-8
- Precedence
 - Configured rule order 34-37
 - Precedence value 34-37

- precedence
 - ACLs 35-6
 - Configured rule order 35-6
 - for policies 34-37, 35-6
 - Precedence value 35-6
 - protocol VLAN rules 8-5
- Q**
- QoS
- application examples 34-31, 34-62
 - ASCII-file-only syntax 34-32
 - configuration overview 34-15
 - defaults 34-12
 - enabled/disabled 34-16
 - interaction with other features 34-5
 - overview 34-3
 - quick steps for creating policies 34-31
 - Specifications 34-2
 - traffic prioritization 34-63
 - qos apply** command 34-59
 - global configuration 34-59
 - policy and port configuration 34-59
 - testing conditions 34-43
 - qos clear log** command 34-21
 - qos** command 34-16
 - qos default bridged disposition** command 34-14, 34-16
 - qos default bridged disposition** command
 - for ACLs 35-7
 - qos default multicast disposition** command 34-14, 34-16
 - qos default routed disposition** command 34-14, 34-16
 - qos default servicing mode** command 34-17, 34-26
 - qos flush** command 34-60
 - compared to **policy server flush** command 33-7
 - qos forward log** command 34-20
 - QoS log
 - cleared 34-21
 - displayed 34-20
 - number of display lines 34-19
 - see also* logged events
 - qos log level** command 34-18, 34-19
 - qos port** command 34-23
 - qos port default 802.1p** command 34-28
 - qos port default dscp** command 34-28
 - qos port q minbw maxbw** command 34-27
 - qos port trusted** command 34-29
 - qos reset** command 34-22
 - qos revert** command 34-60
 - qos stats interval** command 34-22
 - qos trust ports** command 34-29
 - qos user-port** command 35-16
 - Quality of Service
 - see* QoS
 - queues
 - shared 34-23
- used for 802.1X 32-12
- VSA 31-12
- RADIUS authentication servers 31-8
- functional privileges 31-11
 - standard attributes 31-8
 - used for 802.1X 32-6
 - VSA 31-10
- Rapid Spanning Tree Algorithm and Protocol
- see* RSTP
- RDP 28-1, 28-5
- advertisement destination address 28-9
 - advertisement interval 28-9
 - advertisement lifetime 28-10
 - application examples 28-3
 - defaults 28-2
 - disable 28-8
 - enable 28-8
 - example 28-5
 - interface 28-6
 - IP address preference 28-10
 - security 28-7
 - specifications 28-2
 - verify information about 28-11
- RDP interface 28-6
- defaults 28-8
- re-authentication
- 802.1X 32-7
- Redirection Policies 34-67
- Remote Authentication Dial-In User Service
- see* RADIUS authentication servers
- resource threshold limits
- configuring 38-43
- Ring Rapid Spanning Tree Algorithm and Protocol
- see* RRSTP
- RIP 27-1
- application examples 27-3
 - defaults 27-2
 - enabling 27-7
 - forced hold-down timer 27-9
 - garbage timer 27-10
 - hold-down timer 27-10
 - host route 27-11
 - interface 27-7
 - invalid timer 27-10
 - IP 27-4
 - loading 27-6
 - redistribution 27-12
 - security 27-18
 - specifications 27-2
 - unloading 27-6
 - update interval 27-9
 - verification 27-19
 - verify information about 27-19
- RIP interface
- creating 27-7
 - deleting 27-7
 - enabling 27-7
 - metric 27-8
 - password 27-18

- receive option 27-8
 - route tag 27-9
 - send option 27-7
 - RMON
 - application examples 38-11
 - defaults 38-11
 - specifications 38-10
 - RMON events
 - displaying list 38-40
 - displaying specific 38-40
 - RMON probes
 - displaying list 38-37
 - displaying statistics 38-38
 - enabling/disabling 38-36
 - rmon probes** command 38-36
 - RMON tables
 - displaying 38-37
 - route map
 - creating 27-13
 - deleting 27-14
 - enabling/disabling administrative status 27-16
 - redistribution 27-16
 - sequencing 27-14
 - Router Discovery Protocol
 - see* RDP
 - router ID 25-15, 26-14
 - router port
 - IP 25-8
 - router primary address 25-15
 - Routing Information Protocol
 - see* RIP
 - RRSTP 11-38
 - configuration 11-39
 - defaults 11-5
 - RSTP 11-6
 - port connection types 11-35
 - rules
 - see* policies
- S**
- sampling intervals
 - configuring 15-12, 38-45
 - viewing 38-45
 - Secure Socket Layer
 - see* SSL
 - security 28-7
 - Security Violation Mode 3-11
 - restrict** mode 3-11
 - shutdown** mode 3-11
 - Server Load Balancing
 - disabling 11-39
 - enabling 11-39
 - severity level
 - see* switch logging
 - shared queues 34-23
 - show 802.1q** command 22-7, 22-10
 - show amap** command 21-5, 21-7
 - show arp** command 25-12
 - show arp filter** command 25-14, 25-28
 - show bridge rrstp configuration** command 11-39
 - show bridge rrstp ring** command 11-39
 - show gvrp configuration port** command 5-9
 - show health** command 38-46
 - show health interval** command 38-45
 - show health threshold** command 38-13, 38-44
 - show icmp control** command 25-32
 - show icmp statistics** command 25-32
 - show ip config** command 25-16, 25-23
 - show ip interface** command 25-9
 - show ip redist** command 27-16
 - show ip rip** command 27-7
 - show ip rip interface** command 27-7
 - show ip route** command 25-11, 26-13
 - show ip route-map** command 27-13
 - show ipv6 interface** command 26-9
 - show linkagg** command 23-12
 - show linkagg port** command 23-12
 - show lldp remote-system** command 20-4, 20-7
 - show lldp statistics** command 20-4, 20-6
 - show log swlog** command 39-12
 - show policy server long** command 33-6
 - show port mirroring status** command 38-21
 - show port monitoring file** command 38-28
 - show port-security** command 3-3
 - show port-security shutdown** command 3-3
 - show qos log** command 34-20
 - show rmon events** command 38-37
 - show rmon probes** command 38-11, 38-37
 - show spantree** command 11-12
 - show spantree mst region** command 10-15
 - show swlog** command 39-4, 39-10
 - show tcp ports** command 25-33
 - show tcp statistics** command 25-33
 - show uddl configuration** command 18-3
 - show uddl statistics port** command 18-3
 - show udp ports** command 25-33
 - show udp statistics** command 25-33
 - show vlan svlan** command 9-29, 17-12
 - show vlan svlan port-binding** command 9-29, 17-12
 - show vlan svlan port-config** command 9-29, 17-12
 - SNMP
 - attributes for LDAP authentication servers 31-23
 - source learning 2-1
 - application examples 2-3
 - defaults 2-2
 - MAC address table 2-1, 2-5
 - source learning time limit 3-8
 - Spanning Tree
 - specifications 11-3, 12-2
 - Spanning Tree Algorithm and Protocol 11-1, 12-1
 - 1x1 operating mode 4-10, 11-12, 11-14
 - application examples 11-10, 11-40
 - bridge ID 11-8, 11-20
 - Bridge Protocol Data Units 6-11, 11-8, 11-21, 11-22, 11-23
 - bridged ports 11-26
 - designated bridge 11-6

- flat operating mode 4-10, 11-12, 11-13
 - path cost 11-31
 - port connection types 11-35
 - Port ID 11-8
 - port ID 11-30
 - port path cost 11-6
 - port roles 11-7
 - port states 11-7, 11-34
 - root bridge 11-6, 11-21, 11-22, 11-23
 - root path cost 11-6
 - topology 11-6, 11-11
 - Topology Change Notification 11-9
 - Spanning Tree Bridge
 - defaults 11-4, 12-2
 - Spanning Tree bridge parameters
 - 802.1D standard protocol 11-20
 - 802.1s multiple spanning tree protocol 10-1, 11-20
 - 802.1w rapid reconfiguration protocol 11-20
 - automatic VLAN containment 11-25
 - forward delay time 11-23
 - hello time 11-21
 - maximum age time 11-22
 - priority 11-20
 - Spanning Tree Modes 10-11
 - 1x1 mode 10-11
 - flat mode 10-11
 - Spanning Tree Port
 - defaults 11-4
 - Spanning Tree port parameters 11-26
 - connection type 11-35
 - link aggregate ports 11-29, 11-31, 11-32, 11-34, 11-36
 - mode 11-34
 - path cost 11-31
 - priority 11-30
 - specification
 - IPv6 26-2
 - Specifications
 - QoS 34-2
 - specifications
 - 802.1AB 20-2
 - 802.1Q 22-2
 - dynamic link aggregation 24-2
 - Ethernet 1-2
 - Ethernet OAM 15-2, 16-2
 - GVRP 5-2
 - interswitch protocols 21-2
 - IP 25-3
 - Port Mapping 7-2
 - port mirroring 38-3
 - port monitoring 38-5, 38-7
 - RDP 28-2
 - RIP 27-2
 - RMON 38-10
 - Spanning Tree 11-3, 12-2
 - static link aggregation 23-2
 - switch health 38-12
 - switch logging 39-2
 - UDLD 18-2
 - VLAN rules 8-2
 - SSL
 - for LDAP authentication servers 31-27
 - policy servers 33-6
 - static agg agg num** command 23-3, 23-9
 - static link aggregation 23-1
 - adding ports 23-9
 - application examples 23-3, 23-11
 - configuration steps 23-7
 - creating 23-8
 - defaults 23-2
 - deleting 23-8
 - deleting ports 23-9
 - disabling 23-10
 - enabling 23-10
 - group names 23-10
 - groups 23-5, 24-7
 - overview 23-5, 24-7
 - specifications 23-2
 - verify information about 23-12
 - static linkagg admin state** command 23-10
 - static linkagg name** command 23-10
 - static linkagg size** command 23-3, 23-8
 - static MAC addresses 2-5
 - static route
 - IP 25-10, 26-13
 - metric 25-11, 26-13
 - subnet mask 25-10
 - static VLAN port assignment 6-4
 - subnet mask 25-10
 - switch health
 - application examples 38-13
 - defaults 38-13
 - monitoring 38-41
 - specifications 38-12
 - switch health statistics
 - resetting 38-47
 - viewing 38-46
 - switch logging
 - application examples 39-4
 - application ID 39-6
 - defaults 39-3
 - output 39-9
 - severity level 39-8
 - specifications 39-2
 - status 39-10
 - swlog appid level** command 39-6
 - swlog clear** command 39-11
 - swlog** command 39-4, 39-6
 - swlog output** command 34-20
 - swlog output** command 39-9
 - swlog output flash file-size** command 39-11
- ## T
- TCN BPDU
 - see* Topology Change Notification BPDU
 - TCP
 - statistics 25-33

- time-to-live
 - see* TTL
 - Topology Change Notification BPDU 11-9
 - ToS
 - trusted ports 34-28
 - traceroute** command 25-33
 - traffic prioritization 34-63
 - Transparent Switching 5-8
 - trap port link** command 1-7
 - traps
 - port link messages 1-7
 - trusted ports
 - see also* ports
 - used with QoS policies 34-29
 - TTL value 25-16
- U**
- UDLD
 - application examples 18-3
 - defaults 18-2
 - disabling on port 18-6
 - disabling on switch 18-6
 - enabling on port 18-6
 - overview 18-4
 - show 18-8
 - specifications 18-2
 - udld** command 18-3
 - udld port** command 18-3
 - UDP 25-33
 - statistics 25-33
 - User Datagram Protocol
 - see* UDP
 - users
 - functional privileges 31-11, 31-22
- V**
- Vendor Specific Attributes
 - see* VSAs
 - vlan 802.1q** command 4-7, 4-9, 6-4, 22-5
 - vlan 802.1q frame type** command 22-6
 - VLAN advertisements
 - application examples 5-4
 - vlan** command 5-5, 25-4, 27-3
 - vlan dhcp generic** command 8-10
 - vlan dhcp mac** command 8-9
 - vlan dhcp mac range** command 8-9
 - vlan dhcp port** command 8-10
 - vlan ip** command 8-11
 - vlan mac** command 8-10
 - vlan mac range** command 8-11
 - vlan mobile-tag** command 4-9, 6-5
 - vlan port 802.1x** command 32-9
 - vlan port authenticate** command 6-16
 - vlan port** command 8-13
 - and 802.1X ports 32-4
 - vlan port default** command 4-7, 6-4, 25-4, 27-3
 - vlan port default vlan** command 6-16
 - vlan port default vlan restore** command 6-16
 - vlan port mobile** command 4-8, 6-4, 6-10, 6-11
 - vlan protocol** command 8-12
 - vlan router ip** command 25-4
 - VLAN rules 8-1, 8-8
 - application examples 8-3, 8-14
 - defaults 8-2
 - DHCP 8-5, 8-9, 8-10
 - MAC address 8-5, 8-10
 - MAC range 8-11
 - network address 8-5, 8-11
 - port 8-6, 8-13
 - precedence 8-6
 - protocol 8-5, 8-12
 - specifications 8-2
 - types 8-4
 - VLAN Stacking
 - display list of all or range of configured SVLANs 12-21
 - displaying the configuration 9-29
 - vlan stp** command 4-10
 - vlan svlan** command 11-17
 - VLANs 4-1, 4-5, 12-3
 - 802.1Q 22-3
 - administrative status 4-6
 - application examples 4-3, 4-12, 6-3
 - default VLAN 6-1, 6-12
 - defaults 4-2
 - description 4-6
 - IP multinetting 25-7
 - IP router ports 25-8
 - MAC address aging time 2-9
 - mobile tag classification 4-9
 - operational status 4-5
 - port assignment 4-7, 6-1
 - rule classification 4-8
 - secondary VLAN 6-13
 - Spanning Tree status 4-10
 - tagging 22-3
 - VLAN ID 4-5
 - VSAs
 - for LDAP servers 31-22
 - for RADIUS authentication 31-8
 - RADIUS accounting servers 31-12
 - setting up for RADIUS servers 31-10
- W**
- warnings 39-8

