

Part No. 060566-10 Rev. A
December 2018

OmniSwitch AOS Release 6 Network Configuration Guide

6.7.2 R05

Alcatel·Lucent 
Enterprise
www.al-enterprise.com

**This user guide documents release 6.7.2 R05 of the OmniSwitch 6350, 6450.
The functionality described in this guide is subject to change without notice.**

The Alcatel-Lucent name and logo are trademarks of Nokia used under license by ALE. To view other trademarks used by affiliated companies of ALE Holding, visit: www.al-enterprise.com/en/legal/trademarks-copyright. All other trademarks are the property of their respective owners. The information presented is subject to change without notice. Neither ALE Holding nor any of its affiliates assumes any responsibility for inaccuracies contained herein.



26801 West Agoura Road
Calabasas, CA 91301

Service & Support Contact Information

North America: 800-995-2696

Latin America: 877-919-9526

EMEA: +800 00200100 (Toll Free) or +1(650)385-2193

Asia Pacific: +65 6240 8484

Web: <https://businessportal2.alcatel-lucent.com>

Email: ebg_global_supportcenter@al-enterprise.com

Contents

| | | |
|------------------|---|--------|
| | About This Guide | xliii |
| | Supported Platforms | xliii |
| | Who Should Read this Manual? | xliv |
| | When Should I Read this Manual? | xliv |
| | What is in this Manual? | xliv |
| | What is Not in this Manual? | xlv |
| | How is the Information Organized? | xlv |
| | Documentation Roadmap | xlvi |
| | Related Documentation | xlvii |
| | Product Documentation | xlviii |
| | Technical Support | xlviii |
| Chapter 1 | Configuring Ethernet Ports | 1-1 |
| | In This Chapter | 1-1 |
| | Ethernet Specifications | 1-2 |
| | TDR Specifications | 1-3 |
| | Ethernet Port Defaults | 1-3 |
| | Ethernet Ports Overview | 1-4 |
| | OmniSwitch Series Combo Ports | 1-4 |
| | Valid Port Settings on OmniSwitch | 1-5 |
| | 10/100/1000 Crossover Support | 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 |
| | Displaying Storm Control Details | 1-9 |
| | Configuring the Peak Flood Rate Value | 1-10 |
| | Configuring a Port Alias | 1-12 |
| | Configuring Maximum Frame Sizes | 1-12 |
| | Configuring Digital Diagnostic Monitoring (DDM) | 1-12 |
| | Configuring Energy Efficient Ethernet (802.3az) | 1-13 |

| | |
|---|------------|
| Setting Ethernet Parameters for Non-Combo Ports | 1-14 |
| Setting Interface Line Speed | 1-14 |
| Configuring Duplex Mode | 1-14 |
| Configuring Inter-frame Gap Values | 1-15 |
| Configuring Autonegotiation and Crossover Settings | 1-16 |
| Enabling and Disabling Autonegotiation | 1-16 |
| Configuring Crossover Settings | 1-16 |
| Configuring Flow Control on Non-Combo Ports | 1-17 |
| Setting Ethernet Combo Port Parameters | 1-18 |
| Setting Interface Line Speed for Combo Ports | 1-18 |
| Configuring Duplex Mode for Combo Ports | 1-19 |
| Configuring Autonegotiation and Crossover for Combo Ports | 1-20 |
| Enabling and Disabling Autonegotiation for Combo Ports | 1-20 |
| Configuring Crossover Settings for Combo Ports | 1-21 |
| Configuring Flow Control on Combo Ports | 1-22 |
| Monitoring the Inter-stack Connection | 1-23 |
| Clearing the L2 Statistics for Stacking Ports | 1-23 |
| Using TDR Cable Diagnostics | 1-24 |
| Initiating a TDR Cable Diagnostics Test | 1-24 |
| Displaying TDR Test Results | 1-25 |
| Clearing TDR Test Statistics | 1-26 |
| Clearing TDR Extended Test Statistics | 1-26 |
| Interface Violation Recovery | 1-27 |
| Violation Shutdown and Recovery Methods | 1-27 |
| Interface Violation Exceptions | 1-28 |
| Interaction With Other Features | 1-28 |
| Configuring Interface Violation Recovery | 1-29 |
| Configuring the Violation Recovery Time | 1-29 |
| Configuring the Violation Recovery Maximum Attempts | 1-29 |
| Verifying the Interface Violation Recovery Configuration | 1-30 |
| Link Fault Propagation | 1-31 |
| Interaction With Interfaces Violation Recovery | 1-31 |
| Configuring Link Fault Propagation | 1-32 |
| LFP Application Example: Dual-Home Link | 1-33 |
| Verifying Ethernet Port Configuration | 1-34 |
| 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 |
| | Configuring Hash Chain Length | 2-11 |
| | Displaying Source Learning Information | 2-12 |
| 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 | 3-5 |
| | How LPS Authorizes Source MAC Addresses | 3-6 |
| | Dynamic Configuration of Authorized MAC Addresses | 3-7 |
| | Static Configuration of Authorized MAC Addresses | 3-7 |
| | Understanding the LPS Table | 3-8 |
| | Configuring Learned Port Security | 3-9 |
| | Enabling/Disabling Learned Port Security | 3-9 |
| | Configuring a Source Learning Time Limit | 3-10 |
| | Configuring MAC Movement for Pseudo Static MAC | 3-11 |
| | Learning Window Behavior | 3-12 |
| | Configuring Infinite Learning Window | 3-14 |
| | Configuring Automatic Conversion of MAC Addresses | 3-15 |
| | Configuring MAC Movement | 3-15 |
| | Configuring the Number of Bridged MAC Addresses Allowed | 3-16 |
| | Configuring the Trap Threshold for Bridged MAC Addresses | 3-16 |
| | Configuring the Number of Filtered MAC Addresses Allowed | 3-17 |
| | Configuring Authorized MAC Addresses | 3-17 |
| | Configuring an Authorized MAC Address Range | 3-18 |
| | Selecting the Security Violation Mode | 3-20 |
| | Displaying Learned Port Security Information | 3-21 |
| Chapter 4 | Configuring VLANs | 4-1 |
| | In This Chapter | 4-1 |
| | VLAN Specifications | 4-2 |
| | VLAN Defaults | 4-3 |
| | 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 | 4-6 |
| | Changing the Default VLAN Assignment for a Port | 4-7 |
| | Configuring Dynamic VLAN Port Assignment | 4-7 |
| | Configuring VLAN Rule Classification | 4-8 |
| | Enabling/Disabling VLAN Mobile Tag Classification | 4-8 |
| | Enabling/Disabling Spanning Tree for a VLAN | 4-9 |
| | Configuring VLAN Router Interfaces | 4-10 |
| | What is Single MAC Router Mode? | 4-10 |
| | Bridging VLANs Across Multiple Switches | 4-10 |
| | Verifying the VLAN Configuration | 4-12 |
| Chapter 5 | Configuring GVRP | 5-1 |
| | In This Chapter | 5-1 |
| | GVRP Specifications | 5-2 |
| | GVRP Defaults | 5-2 |
| | GARP Overview | 5-2 |
| | 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 | Configuring MVRP | 6-1 |
| | In This Chapter | 6-1 |
| | MVRP Specifications | 6-2 |
| | MVRP Defaults | 6-3 |
| | Quick Steps for Configuring MVRP | 6-4 |
| | MRP Overview | 6-6 |
| | MVRP Overview | 6-6 |

| | |
|---|------------|
| How MVRP Works | 6-7 |
| Interaction With Other Features | 6-9 |
| GVRP | 6-9 |
| STP | 6-9 |
| IPM VLAN | 6-9 |
| Configuring MVRP | 6-10 |
| Enabling MVRP | 6-10 |
| Enabling Transparent Switching | 6-11 |
| Configuring the Maximum Number of VLANs | 6-11 |
| Configuring MVRP Registration | 6-12 |
| Setting MVRP Normal Registration | 6-12 |
| Setting MVRP Fixed Registration | 6-13 |
| Setting MVRP Forbidden Registration | 6-13 |
| Configuring the MVRP Applicant Mode | 6-14 |
| Modifying MVRP Timers | 6-15 |
| Restricting VLAN Registration | 6-16 |
| Restricting Static VLAN Registration | 6-16 |
| Restricting VLAN Advertisement | 6-17 |
| Verifying the MVRP Configuration | 6-18 |
| Chapter 7 Assigning Ports to VLANs | 7-1 |
| In This Chapter | 7-1 |
| Port Assignment Specifications | 7-2 |
| Port Assignment Defaults | 7-3 |
| Sample VLAN Port Assignment | 7-3 |
| Statically Assigning Ports to VLANs | 7-4 |
| Dynamically Assigning Ports to VLANs | 7-5 |
| How Dynamic Port Assignment Works | 7-5 |
| VLAN Mobile Tag Classification | 7-5 |
| VLAN Rule Classification | 7-8 |
| Configuring Dynamic VLAN Port Assignment | 7-10 |
| Enabling/Disabling Port Mobility | 7-11 |
| Ignoring Bridge Protocol Data Units (BPDU) | 7-11 |
| Understanding Mobile Port Properties | 7-12 |
| What is a Configured Default VLAN? | 7-12 |
| What is a Secondary VLAN? | 7-13 |
| Configuring Mobile Port Properties | 7-16 |
| Enable/Disable Default VLAN | 7-16 |
| Enable/Disable Default VLAN Restore | 7-17 |
| Enable/Disable 802.1X Port-Based Access Control | 7-17 |
| Verifying VLAN Port Associations and Mobile Port Properties | 7-18 |
| Understanding ‘show vlan port’ Output | 7-18 |
| Understanding ‘show vlan port mobile’ Output | 7-19 |
| Chapter 8 Configuring Port Mapping | 8-1 |
| In This Chapter | 8-1 |

| | |
|---|------------|
| Port Mapping Specifications | 8-2 |
| Port Mapping Defaults | 8-2 |
| Quick Steps for Configuring Port Mapping | 8-2 |
| Creating/Deleting a Port Mapping Session | 8-3 |
| Creating a Port Mapping Session | 8-3 |
| Deleting a User/Network Port of a Session | 8-3 |
| Deleting a Port Mapping Session | 8-3 |
| Enabling/Disabling a Port Mapping Session | 8-4 |
| Enabling a Port Mapping Session | 8-4 |
| Disabling a Port Mapping Session | 8-4 |
| Configuring a Port Mapping Direction | 8-4 |
| Configuring Unidirectional Port Mapping | 8-4 |
| Restoring Bidirectional Port Mapping | 8-4 |
| Sample Port Mapping Configuration | 8-5 |
| Example Port Mapping Overview | 8-5 |
| Example Port Mapping Configuration Steps | 8-6 |
| Verifying the Port Mapping Configuration | 8-6 |
| Chapter 9 | |
| Defining VLAN Rules | 9-1 |
| In This Chapter | 9-1 |
| VLAN Rules Specifications | 9-2 |
| VLAN Rules Defaults | 9-2 |
| Sample VLAN Rule Configuration | 9-3 |
| VLAN Rules Overview | 9-3 |
| VLAN Rule Types | 9-3 |
| DHCP Rules | 9-5 |
| MAC Address Rules | 9-5 |
| Network Address Rules | 9-5 |
| Protocol Rules | 9-5 |
| Port Rules | 9-6 |
| Understanding VLAN Rule Precedence | 9-6 |
| Configuring VLAN Rule Definitions | 9-8 |
| Defining DHCP MAC Address Rules | 9-9 |
| Defining DHCP MAC Range Rules | 9-9 |
| Defining DHCP Port Rules | 9-10 |
| Defining DHCP Generic Rules | 9-10 |
| Defining MAC Address Rules | 9-10 |
| Defining MAC Range Rules | 9-11 |
| Defining IP Network Address Rules | 9-11 |
| Defining Protocol Rules | 9-12 |
| Defining Port Rules | 9-13 |
| Application Example: DHCP Rules | 9-14 |
| The VLANs | 9-14 |
| DHCP Servers and Clients | 9-15 |

| | |
|--|-------------|
| Verifying VLAN Rule Configuration | 9-17 |
| Chapter 10 | |
| Configuring VLAN Stacking | 10-1 |
| In This Chapter | 10-2 |
| VLAN Stacking Specifications | 10-3 |
| VLAN Stacking Defaults | 10-3 |
| VLAN Stacking Overview | 10-4 |
| How VLAN Stacking Works | 10-6 |
| Traffic Engineering and Translation at UNI and NNI Ports | 10-7 |
| VLAN Stacking Services | 10-9 |
| Interaction With Other Features | 10-10 |
| GARP VLAN Registration Protocol (GVRP) | 10-10 |
| IP Multicast VLANs | 10-10 |
| Link Aggregation | 10-11 |
| Quality of Service (QoS) | 10-11 |
| Ring Rapid Spanning Tree Protocol (RRSTP) | 10-11 |
| Spanning Tree | 10-11 |
| Quick Steps for Configuring VLAN Stacking | 10-12 |
| Configuring VLAN Stacking Services | 10-14 |
| Configuring SVLANs | 10-16 |
| Configuring a VLAN Stacking Service | 10-17 |
| Configuring VLAN Stacking Network Ports | 10-18 |
| Configuring NNI Port Parameters | 10-18 |
| Configuring a VLAN Stacking Service Access Point | 10-20 |
| Configuring VLAN Stacking User Ports | 10-21 |
| Configuring the Type of Customer Traffic to Tunnel | 10-22 |
| Configuring a Service Access Point Profile | 10-23 |
| Configuring SAP profile for Best Effort Service | 10-24 |
| Associating a Profile with a Service Access Point | 10-25 |
| Configuring a UNI Profile | 10-25 |
| Configuring Destination MAC Address | 10-26 |
| Associating UNI Profiles with UNI Ports | 10-26 |
| Configuring Custom L2 Protocol | 10-26 |
| Control Protocol Tunneling Frame Statistics | 10-27 |
| Control HW Tunneling | 10-28 |
| Configuring MAC-Tunneling for SVLAN | 10-29 |
| Global MAC-Tunneling Status | 10-29 |
| SVLAN MAC-Tunneling Configuration | 10-29 |
| VLAN Stacking Application Examples | 10-30 |
| VLAN Stacking Configuration Example | 10-31 |
| Wire-Speed Ethernet Loopback Test | 10-33 |
| Configuring an Ethernet Loopback Test | 10-33 |
| Outward (Egress) Loopback Test | 10-34 |
| Inward (Ingress) Loopback Test | 10-36 |
| Verifying the VLAN Stacking Configuration | 10-37 |

| | | |
|-------------------|--|-------|
| Chapter 11 | Using 802.1Q 2005 Multiple Spanning Tree | 11-1 |
| | In This Chapter | 11-1 |
| | Spanning Tree Specifications | 11-2 |
| | Spanning Tree Bridge Parameter Defaults | 11-2 |
| | Spanning Tree Port Parameter Defaults | 11-3 |
| | Multiple Spanning Tree Region Defaults | 11-3 |
| | MST General Overview | 11-4 |
| | How MSTP Works | 11-4 |
| | Comparing MSTP with STP and RSTP | 11-7 |
| | What is a Multiple Spanning Tree Instance (MSTI) | 11-7 |
| | What is a Multiple Spanning Tree Region | 11-8 |
| | What is the Common Spanning Tree | 11-9 |
| | What is the Internal Spanning Tree (IST) Instance | 11-9 |
| | What is the Common and Internal Spanning Tree Instance | 11-9 |
| | MST Configuration Overview | 11-9 |
| | Using Spanning Tree Configuration Commands | 11-10 |
| | Understanding Spanning Tree Modes | 11-11 |
| | MST Interoperability and Migration | 11-11 |
| | Migrating from Flat Mode STP/RSTP to Flat Mode MSTP | 11-12 |
| | Migrating from 1x1 Mode to Flat Mode MSTP | 11-12 |
| | Quick Steps for Configuring an MST Region | 11-13 |
| | Quick Steps for Configuring MSTIs | 11-15 |
| | Verifying the MST Configuration | 11-18 |
| Chapter 12 | Configuring Spanning Tree | 12-1 |
| | In This Chapter | 12-2 |
| | Spanning Tree Specifications | 12-3 |
| | Spanning Tree Bridge Parameter Defaults | 12-4 |
| | Spanning Tree Port Parameter Defaults | 12-5 |
| | Multiple Spanning Tree (MST) Region Defaults | 12-6 |
| | Ring Rapid Spanning Tree Defaults | 12-7 |
| | Spanning Tree Overview | 12-8 |
| | How the Spanning Tree Topology is Calculated | 12-8 |
| | Bridge Protocol Data Units (BPDU) | 12-10 |
| | Topology Change Notification | 12-11 |
| | Topology Examples | 12-13 |
| | Spanning Tree Operating Modes | 12-15 |
| | Using Flat Spanning Tree Mode | 12-15 |
| | Using 1x1 Spanning Tree Mode | 12-16 |
| | Using 1x1 Spanning Tree Mode with PVST+ | 12-17 |
| | OmniSwitch PVST+ Interoperability | 12-18 |
| | BPDU Processing in PVST+ Mode | 12-19 |

| | |
|---|-------|
| Recommendations and Requirements for PVST+ Configurations | 12-20 |
| Configuring STP Bridge Parameters | 12-21 |
| Bridge Configuration Commands Overview | 12-21 |
| Selecting the Bridge Protocol | 12-24 |
| Configuring the Bridge Priority | 12-24 |
| Configuring the Bridge Hello Time | 12-25 |
| Configuring the Bridge Max Age Time | 12-26 |
| Configuring the Bridge Forward Delay Time | 12-27 |
| Enabling/Disabling the VLAN BPDU Switching Status | 12-28 |
| Configuring the Path Cost Mode | 12-28 |
| Using Automatic VLAN Containment | 12-29 |
| Configuring STP Port Parameters | 12-30 |
| Bridge Configuration Commands Overview | 12-30 |
| Enabling/Disabling Spanning Tree on a Port | 12-33 |
| Spanning Tree on Link Aggregate Ports | 12-33 |
| Configuring Port Priority | 12-34 |
| Port Priority on Link Aggregate Ports | 12-35 |
| Configuring Port Path Cost | 12-35 |
| Path Cost for Link Aggregate Ports | 12-37 |
| Configuring Port Mode | 12-38 |
| Mode for Link Aggregate Ports | 12-38 |
| Configuring Port Connection Type | 12-39 |
| Connection Type on Link Aggregate Ports | 12-40 |
| Configuring Edge Port | 12-40 |
| Restricting Port Roles (Root Guard) | 12-41 |
| Restricting TCN Propagation | 12-41 |
| Limiting BPDU Transmission | 12-41 |
| Using RRSTP | 12-42 |
| Configuring RRSTP | 12-43 |
| Enabling and Disabling RRSTP | 12-43 |
| Creating and Removing RRSTP Rings | 12-43 |
| Sample Spanning Tree Configuration | 12-44 |
| Example Network Overview | 12-44 |
| Example Network Configuration Steps | 12-45 |
| Verifying the Spanning Tree Configuration | 12-47 |
| Chapter 13 Configuring ERP | 13-1 |
| In This Chapter | 13-1 |
| ERPv2 Specifications | 13-2 |
| ERPv2 Defaults | 13-3 |
| ERPv2 Overview | 13-4 |
| ERPv2 Terms | 13-4 |
| ERPv2 Timers | 13-5 |
| ERPv2 Basic Operation | 13-6 |
| R-APS Virtual Channel | 13-7 |
| Revertive / Non-Revertive Mode | 13-7 |
| ERPv2 and RRSTP Differences | 13-8 |

| | |
|---|-------------|
| Interaction With Other Features | 13-9 |
| Spanning Tree | 13-9 |
| VLAN Stacking | 13-9 |
| Ethernet OAM | 13-9 |
| Source Learning | 13-9 |
| QoS Interface | 13-9 |
| MVRP | 13-9 |
| Quick Steps for Configuring ERPV2 with Standard VLANs | 13-10 |
| Quick Steps for Configuring ERPV2 with VLAN Stacking | 13-11 |
| ERPV2 Configuration Overview and Guidelines | 13-12 |
| Configuring an ERP Ring | 13-13 |
| Removing an ERP Ring | 13-14 |
| Configuring an ERPV2 Sub-ring | 13-14 |
| Adding VLANs to Ring | 13-14 |
| Configuring an RPL Port | 13-15 |
| Setting the Wait-to-Restore Timer | 13-15 |
| Setting the Guard Timer | 13-16 |
| Monitoring Remote Ethernet OAM End Points with ERP | 13-16 |
| Configuring ERP with VLAN Stacking NNIs | 13-17 |
| Enabling and Disabling R-APS Virtual Channel | 13-18 |
| Enabling R-APS Virtual Channel | 13-18 |
| Disabling R-APS Virtual Channel | 13-19 |
| Configuring Revertive and Non-revertive Mode | 13-19 |
| Enabling Revertive Mode | 13-19 |
| Non-revertive Mode | 13-20 |
| Clear Non-revertive and Revertive Mode | 13-20 |
| Clearing ERP Statistics | 13-21 |
| ERPV2 Application Example | 13-22 |
| ERPV2 Ring | 13-22 |
| Configuring the Shared Link | 13-23 |
| Configuring the Main RPL Node | 13-23 |
| Configuring the Major Ring | 13-24 |
| Configuring the Sub-ring and RPL Node | 13-24 |
| Configuring the Sub-ring | 13-24 |
| Verifying the ERPV2 Configuration | 13-25 |
| Chapter 14 Configuring Loopback Detection | 14-1 |
| In This Chapter | 14-1 |
| LBD Specifications | 14-2 |
| LBD Defaults | 14-2 |
| Quick Steps for Configuring LBD | 14-3 |
| LBD Overview | 14-4 |
| Transmission Timer | 14-5 |
| Autorecovery | 14-5 |
| Permanent Shutdown for LBD Ports | 14-6 |
| Interaction With Other Features | 14-6 |

| | |
|--|-------|
| Spanning Tree Protocol | 14-6 |
| Link Aggregation | 14-6 |
| Configuring LBD | 14-6 |
| Enabling LBD | 14-7 |
| Enabling LBD on a Port | 14-7 |
| Configuring the LBD Transmission Timer | 14-7 |
| Configuring the Autorecovery Timer | 14-7 |
| Viewing LBD Statistics | 14-7 |
| Recovering a Port from LBD Shutdown | 14-7 |
| LBD Use Case Scenario | 14-8 |
| Verifying the LBD Configuration | 14-10 |
| Chapter 15 | |
| Configuring CPE Test Head | 15-1 |
| In This Chapter | 15-2 |
| CPE Test Head Specifications | 15-2 |
| Quick Steps for Configuring CPE Test Head | 15-3 |
| CPE Test Head Overview | 15-5 |
| CPE Test Head Configuration Overview | 15-6 |
| Configuration Guidelines | 15-6 |
| Configuring a CPE Test Profile | 15-7 |
| Configuring the L2 SAA Test | 15-9 |
| Running a CPE Test | 15-10 |
| Stopping the CPE Test | 15-10 |
| Verifying the CPE Test Configuration and Results | 15-11 |
| Configuring CPE Test Group | 15-13 |
| CPE Test Group Specifications | 15-13 |
| Quick Steps for Configuring CPE Test Group | 15-14 |
| CPE Test Group Overview | 15-17 |
| CPE Test Group Configuration Overview | 15-18 |
| Configuration Guidelines | 15-19 |
| Configuring a CPE Test Group Profile | 15-20 |
| Running a CPE Test Group test | 15-22 |
| Stopping the CPE Test Group test | 15-22 |
| Verifying the CPE Test Group Configuration and Results | 15-23 |
| CPE Test Head Advanced Configuration | 15-25 |
| Running L2 SAA test | 15-25 |
| Configuring Remote Sys MAC | 15-25 |
| Saving the test results on the /flash | 15-26 |
| Sample Test Configurations | 15-27 |
| Sample Unidirectional Test Configuration | 15-27 |
| Sample Bidirectional Test Configuration | 15-28 |

| | | |
|-------------------|--|-------------|
| | Sample Bidirectional Multi-stream Test Configuration | 15-29 |
| Chapter 16 | Configuring PPPoE Intermediate Agent | 16-1 |
| | In This Chapter | 16-1 |
| | PPPoE-IA Specifications | 16-2 |
| | PPPoE-IA Defaults | 16-2 |
| | Quick Steps for Configuring PPPoE-IA | 16-3 |
| | PPPoE Intermediate Agent Overview | 16-5 |
| | How PPPoE-IA Works | 16-5 |
| | Configuring PPPoE-IA | 16-6 |
| | Enabling PPPoE-IA Globally | 16-6 |
| | Enabling PPPoE-IA on a Port | 16-6 |
| | Configuring a Port as Trust or Client | 16-6 |
| | Configuring Access Node Identifier for PPPoE-IA | 16-7 |
| | Configuring Circuit Identifier | 16-7 |
| | Default Circuit ID | 16-7 |
| | ASCII Circuit ID | 16-8 |
| | Configuring Remote Identifier | 16-8 |
| | Verifying PPPoE-IA Configuration | 16-9 |
| Chapter 17 | Configuring Ethernet OAM | 17-1 |
| | In This Chapter | 17-1 |
| | Ethernet OAM Specifications | 17-2 |
| | Ethernet OAM Defaults | 17-2 |
| | Ethernet OAM Overview | 17-3 |
| | Ethernet Service OAM | 17-3 |
| | Elements of Service OAM | 17-3 |
| | CFM Maintenance Domain | 17-4 |
| | MIP CCM Database Support | 17-6 |
| | Performance Monitoring | 17-6 |
| | Interoperability with ITU-T Y.1731 | 17-8 |
| | Quick Steps for Configuring Service OAM | 17-9 |
| | Configuring Ethernet OAM | 17-10 |
| | Configuring a Maintenance Domain | 17-10 |
| | Modifying a Maintenance Domain | 17-11 |
| | Configuring a Maintenance Association | 17-11 |
| | Configuring Maintenance Association Attributes | 17-11 |
| | Configuring a Maintenance End Point | 17-12 |
| | Configuring MEP Attributes | 17-12 |
| | Configuring Loopback | 17-12 |
| | Configuring Linktrace | 17-13 |
| | Configuring the Fault Alarm Time | 17-13 |
| | Configuring the Fault Reset Time | 17-13 |
| | Configuring Ethernet Frame Delay Measurement | 17-13 |
| | CVLANs for EVC MEPs | 17-14 |

| | | |
|-------------------|--|-------------|
| | CVLAN Configuration Overview | 17-15 |
| | Creating MEP on CVLAN | 17-15 |
| | Configuring Remote Fault Propagation (RFP) | 17-15 |
| | Configuring the Allowed CVLAN List | 17-15 |
| | Configuring the CVLAN (ctag) Priority | 17-16 |
| | CVLAN Insertion for Untagged Packets | 17-16 |
| | Viewing the CVLAN Configurations | 17-17 |
| | Verifying the Ethernet OAM Configuration | 17-17 |
| Chapter 18 | Service Assurance Agents (SAA) | 18-1 |
| | In This Chapter | 18-1 |
| | SAA Specifications | 18-2 |
| | SAA Defaults | 18-2 |
| | Quick Steps for Configuring SAA | 18-3 |
| | Configuring Service Assurance Agent (SAA) | 18-3 |
| | Configuring SAA for MAC Addresses | 18-4 |
| | Configuring SAA for IP | 18-4 |
| | Configuring SAA for Ethoam Loopback | 18-4 |
| | Configuring SAA for ETH-DMM | 18-4 |
| | Starting and Stopping SAAs | 18-5 |
| | Enabling Jitter Calculation in SAAs | 18-5 |
| | Displaying the SAA Configuration | 18-6 |
| Chapter 19 | Configuring EFM (LINK OAM) | 19-1 |
| | In This Chapter | 19-1 |
| | LINK OAM Specifications | 19-2 |
| | LINK OAM Defaults | 19-3 |
| | Quick Steps for Configuring LINK OAM | 19-4 |
| | LINK OAM Overview | 19-5 |
| | Discovery | 19-6 |
| | Link Monitoring | 19-6 |
| | Remote Fault detection | 19-6 |
| | Remote Loopback Testing | 19-7 |
| | Interaction With Other Features | 19-7 |
| | Link Aggregate | 19-7 |
| | Connectivity Fault Management | 19-7 |
| | ERP | 19-7 |
| | Configuring LINK OAM | 19-8 |
| | Enabling and Disabling LINK OAM | 19-8 |
| | Setting the Transmit Delay | 19-8 |
| | Enabling and Disabling Propagation of Events | 19-9 |
| | Configuring Link Monitoring | 19-9 |
| | Enabling and Disabling Errored frame period | 19-9 |
| | Enabling and Disabling Errored frame | 19-9 |

| | | |
|-------------------|--|-------|
| | Enabling and Disabling Errored frame seconds summary | 19-10 |
| | Configuring LINK OAM Loopback | 19-10 |
| | Enabling and Disabling Remote loopback | 19-10 |
| | Verifying the LINK OAM Configuration | 19-11 |
| Chapter 20 | Configuring UDLD | 20-1 |
| | In This Chapter | 20-1 |
| | UDLD Specifications | 20-2 |
| | UDLD Defaults | 20-2 |
| | Quick Steps for Configuring UDLD | 20-3 |
| | UDLD Overview | 20-4 |
| | UDLD Operational Mode | 20-4 |
| | Normal Mode | 20-4 |
| | Aggressive Mode | 20-4 |
| | Mechanisms to Detect Unidirectional Links | 20-5 |
| | Neighbor Database Maintenance | 20-5 |
| | Echo Detection | 20-5 |
| | Configuring UDLD | 20-6 |
| | Enabling and Disabling UDLD | 20-6 |
| | Enabling UDLD on a Switch | 20-6 |
| | Disabling UDLD on a Switch | 20-6 |
| | Enabling UDLD on a Port | 20-6 |
| | Disabling UDLD on a Port | 20-6 |
| | Configuring Mode | 20-7 |
| | Configuring Probe-timer | 20-7 |
| | Configuring Echo-wait-timer | 20-7 |
| | Clearing UDLD Statistics | 20-8 |
| | Recovering a Port from UDLD Shutdown | 20-8 |
| | Displaying UDLD Information | 20-8 |
| Chapter 21 | Configuring MAC Retention | 21-1 |
| | In This Chapter | 21-1 |
| | MAC Retention Defaults | 21-1 |
| | MAC Retention Overview | 21-2 |
| | How MAC Retention Works | 21-3 |
| | MAC Retention After Multiple Take-Overs | 21-4 |
| | Configuring MAC Retention | 21-4 |
| | Enabling MAC Retention | 21-4 |
| | Detecting a Duplicate MAC Address | 21-4 |
| | Configuring MAC Release | 21-5 |
| | MAC Retention Applications | 21-5 |
| | Software Failure | 21-6 |
| | Link Failure | 21-7 |

| | | |
|-------------------|---|-------|
| Chapter 22 | Configuring 802.1AB | 22-1 |
| | In This Chapter | 22-1 |
| | 802.1AB Specifications | 22-2 |
| | 802.1AB Defaults Table | 22-2 |
| | Quick Steps for Configuring 802.1AB | 22-4 |
| | Quick Steps for Configuring LLDP-MED Network Policy | 22-5 |
| | LLDP-MED Network Policy for Fixed Ports | 22-5 |
| | LLDP on Mobile Ports | 22-6 |
| | LLDP-MED Network Policy on 802.1x Ports | 22-6 |
| | 802.1AB Overview | 22-8 |
| | LLDP PoE Power Negotiation | 22-9 |
| | mac-phy TLV | 22-9 |
| | power-via-mdi TLV | 22-9 |
| | LLDP-Media Endpoint Devices | 22-10 |
| | LLDP-MED Network Policy | 22-10 |
| | LLDP-MED Network Policy for VLAN Advertisement | 22-11 |
| | Fast Restart of LLDP on Detection of MED | 22-11 |
| | LLDP-MED for IP Phones | 22-11 |
| | LLDP Agent Operation | 22-12 |
| | LLDPDU Transmission and Reception | 22-12 |
| | Aging Time | 22-12 |
| | LLDP Agent Security Mechanism | 22-13 |
| | Nearest Bridge/Edge Mode | 22-14 |
| | Nearest-Edge Mode Operation | 22-15 |
| | Configuring 802.1AB | 22-16 |
| | Configuring LLDPDU Flow | 22-16 |
| | Enabling and Disabling Notification | 22-16 |
| | Enabling and Disabling Management TLV | 22-17 |
| | Enabling and Disabling 802.1 TLV | 22-17 |
| | Enabling and Disabling 802.3 TLV | 22-18 |
| | Enabling and Disabling MED TLV | 22-18 |
| | Enabling and Disabling Proprietary TLV | 22-19 |
| | Setting the Transmit Interval | 22-21 |
| | Setting the Transmit Hold Multiplier Value | 22-21 |
| | Setting the Transmit Delay | 22-21 |
| | Setting the Transmit Fast Start Count | 22-21 |
| | Setting the Reinit Delay | 22-21 |
| | Setting the Notification Interval | 22-21 |
| | Configuring LLDP Security Mechanism | 22-22 |
| | Application Example - LLDP MED | 22-23 |
| | Verifying 802.1AB Configuration | 22-24 |
| Chapter 23 | Using Interswitch Protocols | 23-1 |
| | In This Chapter | 23-1 |
| | AIP Specifications | 23-2 |

| | |
|--|-------------|
| AMAP Defaults | 23-2 |
| AMAP Overview | 23-3 |
| AMAP Transmission States | 23-3 |
| Discovery Transmission State | 23-4 |
| Common Transmission State | 23-4 |
| Passive Reception State | 23-4 |
| Common Transmission and Remote Switches | 23-5 |
| Configuring AMAP | 23-5 |
| Enabling or Disabling AMAP | 23-5 |
| Configuring the AMAP Discovery Time-out Interval | 23-5 |
| Configuring the AMAP Common Time-out Interval | 23-6 |
| Displaying AMAP Information | 23-7 |
| Chapter 24 Configuring 802.1Q | 24-1 |
| In this Chapter | 24-1 |
| 802.1Q Specifications | 24-2 |
| 802.1Q Defaults Table | 24-2 |
| 802.1Q Overview | 24-3 |
| Configuring an 802.1Q VLAN | 24-5 |
| Enabling Tagging on a Port | 24-5 |
| Enabling Tagging with Link Aggregation | 24-5 |
| Configuring the Frame Type | 24-6 |
| Show 802.1Q Information | 24-7 |
| Application Example | 24-7 |
| Verifying 802.1Q Configuration | 24-10 |
| Chapter 25 Configuring Static Link Aggregation | 25-1 |
| In This Chapter | 25-1 |
| Static Link Aggregation Specifications | 25-2 |
| Static Link Aggregation Default Values | 25-2 |
| Quick Steps for Configuring Static Link Aggregation | 25-3 |
| Static Link Aggregation Overview | 25-5 |
| Static Link Aggregation Operation | 25-5 |
| Relationship to Other Features | 25-6 |
| Configuring Static Link Aggregation Groups | 25-7 |
| Configuring Mandatory Static Link Aggregate Parameters | 25-7 |
| Creating and Deleting a Static Link Aggregate Group | 25-8 |
| Creating a Static Aggregate Group | 25-8 |
| Deleting a Static Aggregate Group | 25-8 |
| Adding and Deleting Ports in a Static Aggregate Group | 25-9 |
| Adding Ports to a Static Aggregate Group | 25-9 |
| Removing Ports from a Static Aggregate Group | 25-9 |
| Modifying Static Aggregation Group Parameters | 25-10 |
| Modifying the Static Aggregate Group Name | 25-10 |

| | |
|---|-------|
| Creating a Static Aggregate Group Name | 25-10 |
| Deleting a Static Aggregate Group Name | 25-10 |
| Modifying the Static Aggregate Group Administrative State | 25-10 |
| Enabling the Static Aggregate Group Administrative State | 25-10 |
| Disabling the Static Aggregate Group Administrative State | 25-10 |
| Application Example | 25-11 |
| Displaying Static Link Aggregation Configuration and Statistics | 25-12 |
| Chapter 26 | |
| Configuring Dynamic Link Aggregation | 26-1 |
| In This Chapter | 26-1 |
| Dynamic Link Aggregation Specifications | 26-2 |
| Dynamic Link Aggregation Default Values | 26-3 |
| Quick Steps for Configuring Dynamic Link Aggregation | 26-4 |
| Dynamic Link Aggregation Overview | 26-6 |
| Dynamic Link Aggregation Operation | 26-6 |
| Relationship to Other Features | 26-8 |
| Configuring Dynamic Link Aggregate Groups | 26-9 |
| Configuring Mandatory Dynamic Link Aggregate Parameters | 26-9 |
| Creating and Deleting a Dynamic Aggregate Group | 26-10 |
| Creating a Dynamic Aggregate Group | 26-10 |
| Deleting a Dynamic Aggregate Group | 26-10 |
| Configuring Ports to Join and Removing Ports in a Dynamic Aggregate Group | 26-11 |
| Configuring Ports To Join a Dynamic Aggregate Group | 26-11 |
| Removing Ports from a Dynamic Aggregate Group | 26-12 |
| Modifying Dynamic Link Aggregate Group Parameters | 26-13 |
| Modifying Dynamic Aggregate Group Parameters | 26-13 |
| Modifying the Dynamic Aggregate Group Name | 26-14 |
| Modifying the Dynamic Aggregate Group Administrative State | 26-14 |
| Configuring and Deleting the Dynamic Aggregate Group Actor Administrative Key | 26-15 |
| Modifying the Dynamic Aggregate Group Actor System Priority | 26-15 |
| Modifying the Dynamic Aggregate Group Actor System ID | 26-16 |
| Modifying the Dynamic Aggregate Group Partner Administrative Key | 26-16 |
| Modifying the Dynamic Aggregate Group Partner System Priority | 26-17 |
| Modifying the Dynamic Aggregate Group Partner System ID | 26-17 |
| Modifying Dynamic Link Aggregate Actor Port Parameters | 26-18 |
| Modifying the Actor Port System Administrative State | 26-18 |
| Modifying the Actor Port System ID | 26-20 |
| Modifying the Actor Port System Priority | 26-20 |
| Modifying the Actor Port Priority | 26-21 |
| Modifying Dynamic Aggregate Partner Port Parameters | 26-22 |
| Modifying the Partner Port System Administrative State | 26-22 |
| Modifying the Partner Port Administrative Key | 26-24 |
| Modifying the Partner Port System ID | 26-24 |
| Modifying the Partner Port System Priority | 26-25 |
| Modifying the Partner Port Administrative Status | 26-26 |
| Modifying the Partner Port Priority | 26-26 |

| | | |
|-------------------|--|-------------|
| | Edge Feature - LACP WTR Delay on Bootup | 26-27 |
| | Application Examples | 26-29 |
| | Dynamic Link Aggregation Example | 26-29 |
| | Link Aggregation and Spanning Tree Example | 26-30 |
| | Link Aggregation and QoS Example | 26-31 |
| | Displaying Dynamic Link Aggregation Configuration and Statistics | 26-33 |
| Chapter 27 | Configuring Dual-Home Links | 27-1 |
| | In This Chapter | 27-2 |
| | Dual-Home Link Aggregation Specifications | 27-3 |
| | Dual-Home Link Active-Active Defaults | 27-4 |
| | Dual-Home Link Active-Active | 27-5 |
| | DHL Active-Active Operation | 27-5 |
| | Protected VLANs | 27-6 |
| | DHL Port Types | 27-6 |
| | DHL Pre-Emption Timer | 27-6 |
| | MAC Address Flushing | 27-6 |
| | DHL Configuration Guidelines | 27-8 |
| | Configuring DHL Active-Active | 27-8 |
| | Dual-Home Link Active-Active Example | 27-10 |
| | CLI Command Sequence Example | 27-11 |
| | Recommended DHL Active-Active Topology | 27-12 |
| | Unsupported DHL Active-Active Topology (Network Loops) | 27-13 |
| | Displaying the Dual-Home Link Configuration | 27-14 |
| Chapter 28 | Configuring IP | 28-1 |
| | In This Chapter | 28-1 |
| | IP Specifications | 28-2 |
| | IP Defaults | 28-4 |
| | Quick Steps for Configuring IP Forwarding | 28-4 |
| | IP Overview | 28-5 |
| | IP Protocols | 28-5 |
| | Transport Protocols | 28-5 |
| | Application-Layer Protocols | 28-5 |
| | Additional IP Protocols | 28-6 |
| | IP Forwarding | 28-6 |
| | Configuring an IP Router Interface | 28-7 |
| | Modifying an IP Router Interface | 28-9 |
| | Removing an IP Router Interface | 28-9 |
| | Configuring a Loopback0 Interface | 28-10 |
| | Loopback0 Address Advertisement | 28-10 |
| | Creating a Static Route | 28-10 |
| | Creating a Default Route | 28-12 |
| | Configuring Address Resolution Protocol (ARP) | 28-13 |
| | Adding a Permanent Entry to the ARP Table | 28-13 |

| | |
|--|-------|
| Deleting a Permanent Entry from the ARP Table | 28-14 |
| Clearing a Dynamic Entry from the ARP Table | 28-14 |
| Local Proxy ARP | 28-14 |
| Dynamic Proxy ARP - Mac Forced Forwarding | 28-15 |
| ARP Filtering | 28-17 |
| IP Configuration | 28-18 |
| Configuring the DHCP Client Interface | 28-18 |
| Configuring the Router Primary Address | 28-18 |
| Configuring the Router ID | 28-18 |
| Configuring the Route Preference of a Router | 28-18 |
| Configuring the Time-to-Live (TTL) Value | 28-19 |
| Configuring Route Map Redistribution | 28-19 |
| Using Route Maps | 28-20 |
| Configuring Route Map Redistribution | 28-24 |
| Route Map Redistribution Example | 28-25 |
| IP-Directed Broadcasts | 28-26 |
| Controlled Directed Broadcasts | 28-26 |
| Denial of Service (DoS) Filtering | 28-27 |
| Enabling/Disabling IP Services | 28-32 |
| Extend IPv4 Interfaces and Static Routes Support on OmniSwitch | 28-33 |
| Managing IP | 28-34 |
| Internet Control Message Protocol (ICMP) | 28-34 |
| ICMP Control Table | 28-37 |
| ICMP Statistics Table | 28-37 |
| Using the Ping Command | 28-37 |
| Tracing an IP Route | 28-38 |
| Displaying TCP Information | 28-38 |
| Displaying UDP Information | 28-38 |
| Displaying Probe Information | 28-38 |
| Two-Way Active Measurement Protocol (TWAMP) | 28-39 |
| TWAMP on OmniSwitch | 28-39 |
| TWAMP Operation | 28-40 |
| TWAMP Metrics Measurement | 28-40 |
| Configuring TWAMP Server on the Switch | 28-41 |
| Removing the TWAMP Server Configuration from the Switch | 28-41 |
| Viewing the TWAMP Server Information and Established Client Connections | 28-41 |
| Network Address Translation | 28-42 |
| Configuring NAT | 28-42 |
| Verifying the IP Configuration | 28-44 |
| Chapter 29 | |
| Configuring IPv6 | 29-1 |
| In This Chapter | 29-1 |
| IPv6 Specifications | 29-2 |
| IPv6 Defaults | 29-3 |
| Quick Steps for Configuring IPv6 Routing | 29-3 |
| IPv6 Overview | 29-4 |

| | |
|---|-------|
| IPv6 Addressing | 29-5 |
| IPv6 Address Notation | 29-6 |
| IPv6 Address Prefix Notation | 29-6 |
| Auto Configuration of IPv6 Addresses | 29-7 |
| Router Advertisement (RA) Filtering | 29-8 |
| Configuring an IPv6 Interface | 29-9 |
| Modifying an IPv6 Interface | 29-10 |
| Removing an IPv6 Interface | 29-10 |
| Assigning IPv6 Addresses | 29-11 |
| Removing an IPv6 Address | 29-12 |
| Creating an IPv6 Static Route | 29-12 |
| Configuring the Route Preference of a Router | 29-13 |
| Configuring Route Map Redistribution | 29-13 |
| Using Route Maps | 29-14 |
| Configuring Route Map Redistribution | 29-18 |
| Route Map Redistribution Example | 29-19 |
| Configuring Router Advertisement (RA) Filtering | 29-20 |
| Verifying the IPv6 Configuration | 29-21 |
| Chapter 30 | |
| Configuring RIP | 30-1 |
| In This Chapter | 30-1 |
| RIP Specifications | 30-2 |
| RIP Defaults | 30-2 |
| Quick Steps for Configuring RIP Routing | 30-3 |
| RIP Overview | 30-4 |
| RIP Version 2 | 30-5 |
| RIP Routing | 30-6 |
| Loading RIP | 30-7 |
| Enabling RIP | 30-7 |
| Creating a RIP Interface | 30-7 |
| Enabling a RIP Interface | 30-8 |
| Configuring the RIP Interface Send Option | 30-8 |
| Configuring the RIP Interface Receive Option | 30-8 |
| Configuring the RIP Interface Metric | 30-9 |
| Configuring the RIP Interface Route Tag | 30-9 |
| RIP Options | 30-10 |
| Configuring the RIP Forced Hold-Down Interval | 30-10 |
| Configuring the RIP Update Interval | 30-10 |
| Configuring the RIP Invalid Timer | 30-10 |
| Configuring the RIP Garbage Timer | 30-11 |
| Configuring the RIP Hold-Down Timer | 30-11 |
| Reducing the Frequency of RIP Routing Updates | 30-11 |
| Enabling a RIP Host Route | 30-11 |
| Configuring Redistribution | 30-12 |

| | | |
|-------------------|---|-------|
| | Using Route Maps | 30-12 |
| | Configuring Route Map Redistribution | 30-16 |
| | Route Map Redistribution Example | 30-17 |
| | RIP Security | 30-18 |
| | Configuring Authentication Type | 30-18 |
| | Configuring Passwords | 30-18 |
| | Verifying the RIP Configuration | 30-19 |
| Chapter 31 | Configuring RDP | 31-1 |
| | In This Chapter | 31-1 |
| | RDP Specifications | 31-2 |
| | RDP Defaults | 31-2 |
| | Quick Steps for Configuring RDP | 31-3 |
| | RDP Overview | 31-5 |
| | RDP Interfaces | 31-6 |
| | Security Concerns | 31-7 |
| | Enabling/Disabling RDP | 31-8 |
| | Creating an RDP Interface | 31-8 |
| | Specifying an Advertisement Destination Address | 31-9 |
| | Defining the Advertisement Interval | 31-9 |
| | Setting the Maximum Advertisement Interval | 31-9 |
| | Setting the Minimum Advertisement Interval | 31-10 |
| | Setting the Advertisement Lifetime | 31-10 |
| | Setting the Preference Levels for Router IP Addresses | 31-10 |
| | Verifying the RDP Configuration | 31-11 |
| Chapter 32 | Configuring DHCP | 32-1 |
| | In This Chapter | 32-1 |
| | DHCP Relay Specifications | 32-3 |
| | DHCPv6 Relay Specifications | 32-4 |
| | DHCP Relay Defaults | 32-5 |
| | DHCPv6 Relay Defaults | 32-6 |
| | Quick Steps for Setting Up DHCP Relay | 32-7 |
| | Quick Steps for Setting Up DHCPv6 Relay | 32-8 |
| | DHCP Relay Overview | 32-9 |
| | DHCP | 32-10 |
| | DHCP and the OmniSwitch | 32-10 |
| | External DHCP Relay Application | 32-11 |
| | Internal DHCP Relay | 32-12 |
| | DHCP Relay Implementation | 32-13 |
| | Global DHCP | 32-14 |
| | Setting the IP Address | 32-14 |

| | |
|---|-------|
| Per-VLAN DHCP | 32-15 |
| Identifying the VLAN | 32-15 |
| Configuring BOOTP/DHCP Relay Parameters | 32-15 |
| Setting the Forward Delay | 32-16 |
| Setting Maximum Hops | 32-16 |
| Setting the Relay Forwarding Option | 32-16 |
| Configuring the DHCP Client Interface | 32-17 |
| Configuring the DHCP Client Interface | 32-17 |
| DHCP Option-12 and DHCP Option-2 | 32-18 |
| Reload and Takeover | 32-18 |
| DHCP Client Interface Guidelines | 32-19 |
| DHCP Server Preference in DHCP Client Interface | 32-19 |
| Vendor Class Identifier and Preference to OXO DHCP Server | 32-21 |
| Configuring UDP Port Relay | 32-22 |
| Enabling/Disabling UDP Port Relay | 32-23 |
| Specifying a Forwarding VLAN | 32-23 |
| Configuring DHCP Security Features | 32-24 |
| Using the Relay Agent Information Option (Option-82) | 32-25 |
| How the Relay Agent Processes DHCP Packets from the Client | 32-26 |
| How the Relay Agent Processes DHCP Packets from the Server | 32-26 |
| Enabling the Relay Agent Information Option-82 | 32-27 |
| Configuring a Relay Agent Information Option-82 Policy | 32-27 |
| Using DHCP Snooping | 32-28 |
| DHCP Snooping Configuration Guidelines | 32-29 |
| Enabling DHCP Snooping | 32-29 |
| Configuring the Port Trust Mode | 32-32 |
| Bypassing the Option-82 Check on Untrusted Ports | 32-32 |
| Configuring Port IP Source Filtering | 32-33 |
| Configuring IP Source Filtering | 32-34 |
| Configuring the DHCP Snooping Binding Table | 32-36 |
| DHCP Snooping ISF ARP-Allow | 32-37 |
| Layer 2 DHCP Snooping | 32-38 |
| DHCP Snooping Global Mode Settings | 32-38 |
| Verifying the DHCP Relay Configuration | 32-41 |
| DHCPv6 Relay Overview | 32-42 |
| Configuring DHCPv6 Relay | 32-43 |
| Layer 3 DHCPv6 relay | 32-43 |
| Layer 2 DHCPv6 relay or Lightweight DHCPv6 Relay Agent (LDRA) | 32-43 |
| Global DHCPv6 | 32-44 |
| Setting the IPv6 Address | 32-44 |
| Per-VLAN DHCPv6 | 32-44 |
| Identifying the VLAN | 32-44 |
| Configuring DHCPv6 Relay Parameters | 32-45 |
| Setting Maximum Hops | 32-45 |
| Setting the DHCPv6 Relay Forwarding Option | 32-45 |
| Using the DHCPv6 Relay Agent Information | 32-46 |
| Configuring Interface ID | 32-46 |
| Configuring Remote ID | 32-46 |

| | | |
|-------------------|--|-------------|
| | VRF Support | 32-47 |
| | DHCPv6 Snooping Configuration Guidelines | 32-47 |
| | Enabling DHCPv6 Snooping | 32-47 |
| | Configuring the Trust Mode for Ports and Link Aggregates | 32-48 |
| | Configuring the DHCPv6 Snooping Binding Table | 32-49 |
| | Configuring the Binding Table Timeout | 32-49 |
| | Synchronizing the Binding Table | 32-49 |
| | Binding Table Retention | 32-50 |
| | Configuring IPv6 Source Filtering | 32-50 |
| | Configuring Port IPv6 Source Filtering | 32-50 |
| | Configuring VLAN IPv6 Source Filtering | 32-51 |
| | Verifying the DHCPv6 Relay Configuration | 32-52 |
| Chapter 33 | Configuring DHCP Server | 33-1 |
| | In This Chapter | 33-1 |
| | DHCP Server Specifications | 33-2 |
| | DHCP Server Default Values | 33-2 |
| | Quick Steps to Configure Internal DHCP Server | 33-3 |
| | DHCP Server Overview | 33-5 |
| | The DHCP process | 33-5 |
| | Internal DHCP Server on OmniSwitch | 33-6 |
| | Interaction With Other Features | 33-6 |
| | Virtual Router Forwarding (VRF) | 33-6 |
| | BootP/UDP Relay | 33-6 |
| | DHCP Snooping | 33-6 |
| | IP Interfaces | 33-6 |
| | Configuring DHCP Server on OmniSwitch | 33-7 |
| | DHCP Template files | 33-7 |
| | Policy file | 33-7 |
| | DHCP Configuration Files | 33-8 |
| | dhcpd.conf File | 33-8 |
| | dhcpd.conf.lastgood File | 33-9 |
| | DHCP Server Database file | 33-9 |
| | DHCP Server Application Example | 33-10 |
| | Verifying DHCP Server Configuration | 33-12 |
| | Configuration File Parameters and Syntax | 33-13 |
| | Policy File Parameters and Syntax | 33-26 |
| Chapter 34 | Configuring VRRP | 34-1 |
| | In This Chapter | 34-1 |
| | VRRP Specifications | 34-3 |
| | VRRP Defaults | 34-3 |
| | Quick Steps for Creating a Virtual Router | 34-5 |

| | |
|---|-------------|
| VRRP Overview | 34-6 |
| Why Use VRRP? | 34-7 |
| Definition of a Virtual Router | 34-7 |
| VRRP MAC Addresses | 34-8 |
| ARP Requests | 34-8 |
| ICMP Redirects | 34-8 |
| VRRP Startup Delay | 34-9 |
| VRRP Tracking | 34-9 |
| Configuring Collective Management Functionality | 34-9 |
| Interaction With Other Features | 34-9 |
| VRRP Configuration Overview | 34-10 |
| Basic Virtual Router Configuration | 34-10 |
| Creating/Deleting a Virtual Router | 34-10 |
| Specifying an IP Address for a Virtual Router | 34-11 |
| Configuring the Advertisement Interval | 34-12 |
| Configuring Virtual Router Priority | 34-12 |
| Setting Preemption for Virtual Routers | 34-13 |
| Configuring VRRP Authentication | 34-13 |
| Enabling/Disabling a Virtual Router | 34-14 |
| Setting VRRP Traps | 34-15 |
| Setting VRRP Startup Delay | 34-15 |
| Configuring Collective Management Functionality | 34-16 |
| Changing Default Parameter Values for all Virtual Routers | 34-16 |
| Changing Default Parameter Values for a Virtual Router Group | 34-17 |
| Verifying the VRRP Configuration | 34-19 |
| VRRPv3 Configuration Overview | 34-20 |
| Basic VRRPv3 Virtual Router Configuration | 34-20 |
| Creating/Deleting a VRRPv3 Virtual Router | 34-20 |
| Specify an IPv6 Address for a VRRPv3 Virtual Router | 34-22 |
| Configuring the VRRPv3 Advertisement Interval | 34-22 |
| Configuring the VRRPv3 Virtual Router Priority | 34-23 |
| Setting Preemption for VRRPv3 Virtual Routers | 34-23 |
| Configuring VRRP Authentication | 34-24 |
| Enabling/Disabling a VRRPv3 Virtual Router | 34-25 |
| Setting VRRPv3 Traps | 34-25 |
| Verifying the VRRPv3 Configuration | 34-26 |
| Creating Tracking Policies | 34-27 |
| Associating a Tracking Policy with a VRRPv2/VRRPv3 Virtual Router | 34-27 |
| VRRP Application Example | 34-28 |
| VRRP Tracking Example | 34-30 |
| VRRPv3 Application Example | 34-32 |
| VRRPv3 Tracking Example | 34-33 |
| Chapter 35 Configuring Access Guardian | 35-1 |
| In This Chapter | 35-2 |
| Access Guardian Specifications | 35-4 |

| | |
|---|-------|
| Access Guardian Defaults | 35-5 |
| Quick Steps for Configuring Access Guardian | 35-7 |
| Quick Steps for Configuring User Network Profiles | 35-9 |
| Quick Steps for Configuring User Network Profile Mobile Rules | 35-10 |
| Quick Steps for Configuring Host Integrity Check | 35-12 |
| Access Guardian Overview | 35-14 |
| Authentication and Classification | 35-15 |
| Control Over Access Guardian Authentication (802.1x Bypass) | 35-16 |
| Captive Portal Bypass | 35-16 |
| Using Device Classification Policies | 35-16 |
| Host Integrity Check (End-User Compliance) | 35-19 |
| How it Works | 35-19 |
| HIC Server Redundancy and Failure Mode | 35-20 |
| User Network Profiles (Role-Based Access) | 35-22 |
| What are UNP Mobile Rules? | 35-23 |
| Dynamic UNP | 35-24 |
| Dynamic UNP Operation Summary Table | 35-24 |
| Interaction With Other Features | 35-25 |
| Quality of Service (QoS) | 35-25 |
| Host Integrity Check - InfoExpress | 35-25 |
| Captive Portal - Browser Support | 35-26 |
| Setting Up Port-Based Network Access Control | 35-26 |
| Setting 802.1X Switch Parameters | 35-26 |
| Enabling MAC Authentication | 35-26 |
| MAC accounting | 35-27 |
| Enabling an Authentication Server Down Policy | 35-27 |
| Critical Voice VLAN | 35-28 |
| MAC Verification to Classify IP Phone Traffic | 35-28 |
| Configuring Critical Voice VLAN | 35-29 |
| Removing the Critical Voice VLAN Configuration | 35-29 |
| Verifying the Critical Voice VLAN Configuration | 35-29 |
| Enabling 802.1X on Ports | 35-30 |
| Configuring 802.1X Port Parameters | 35-30 |
| Configuring Access Guardian Policies | 35-31 |
| Configuring Supplicant Policies | 35-32 |
| Supplicant Policy Examples | 35-33 |
| Configuring Non-supplicant Policies | 35-34 |
| Non-supplicant Policy Examples | 35-36 |
| Configuring the Captive Portal Policy | 35-38 |
| Configuring 802.1x Authentication Bypass | 35-40 |
| Configuration Guidelines | 35-40 |
| Example: Supplicant Bypass with allow-eap as Fail | 35-41 |
| Configuring Captive Portal Authentication | 35-42 |
| Configuring Captive Portal Session Parameters | 35-43 |
| Customizing Captive Portal | 35-43 |
| Authenticating with Captive Portal | 35-45 |
| Logging Into the Network with Captive Portal | 35-45 |
| Logging Off the Network with Captive Portal | 35-48 |

| | |
|---|-------|
| Configuring Host Integrity Check | 35-49 |
| Configuring HIC Redundancy | 35-50 |
| Configuring User Network Profiles | 35-51 |
| Configuring QoS Policy Lists | 35-51 |
| Port Bandwidth Through RADIUS | 35-52 |
| Configuring Bandwidth Profiling on a UNP | 35-52 |
| Multiple User Authentication on the Same Port | 35-53 |
| Configuring User Network Profile Mobile Rules | 35-54 |
| Configuring Dynamic UNP | 35-54 |
| OmniAccess Stellar AP Integration | 35-55 |
| How it Works | 35-55 |
| Configuration Guidelines | 35-56 |
| OmniAccess Stellar AP Configuration Guidelines | 35-58 |
| Quick Steps for Configuring OmniSwitch AP Discovery | 35-58 |
| Verify the OmniSwitch Configuration | 35-59 |
| Verifying Access Guardian Users | 35-62 |
| Logging Users out of the Network | 35-64 |
| Verifying the Access Guardian Configuration | 35-65 |
| Bring Your Own Device (BYOD) Overview | 35-66 |
| Key Components of a BYOD Solution | 35-67 |
| ClearPass Policy Manager | 35-68 |
| OmniSwitch Integration with UPAM or CPPM for BYOD Support | 35-69 |
| Port Bounce | 35-73 |
| Pause timer | 35-73 |
| Configuring OmniSwitch BYOD Support | 35-74 |
| Configuring the UPAM or CPPM server as an AAA RADIUS Server | 35-74 |
| Configuring 802.1x | 35-74 |
| Configuring Redirection with Dynamic URLs | 35-74 |
| Configuring UNP Profiles | 35-74 |
| Configuring Port Bounce | 35-75 |
| Configuring the Pause Timer | 35-75 |
| BYOD Authentication Process Overview | 35-75 |
| Authentication for Registered Devices (802.1x) | 35-75 |
| Authentication for Network Devices (MAC Authentication) | 35-75 |
| Authentication for Guest Devices and Employee On-boarding | 35-76 |
| Zero Configuration Networking (mDNS and SSDP) | 35-77 |
| Quick Steps for Zero Configuration | 35-78 |
| mDNS Work Flow | 35-81 |
| Operating Principle | 35-81 |
| Backward Compatibility | 35-84 |
| Verifying the Zero Configuration | 35-85 |
| BYOD Application Examples | 35-86 |
| Employee Registered Device - 802.1x Authentication | 35-86 |
| IP Phone - MAC Authentication | 35-86 |
| Guest Device - MAC Authentication with Guest Login | 35-86 |
| Application Example 1 (802.1x) - OmniSwitch Configuration | 35-87 |
| Application Example 1 (802.1x) - ClearPass Configuration | 35-88 |
| Application Example 2 (IP Phone) - OmniSwitch Configuration | 35-94 |
| Application Example 2 (IP Phone) - ClearPass Configuration | 35-95 |

| | |
|--|-------------|
| Application Example 3 (Guest) - OmniSwitch Configuration | 35-98 |
| Application Example 3 (Guest) - ClearPass Configuration | 35-99 |
| Verifying BYOD Configuration | 35-104 |
| Chapter 36 | |
| Managing Authentication Servers | 36-1 |
| In This Chapter | 36-1 |
| Authentication Server Specifications | 36-2 |
| Server Defaults | 36-3 |
| RADIUS Authentication Servers | 36-3 |
| TACACS+ Authentication Servers | 36-3 |
| LDAP Authentication Servers | 36-4 |
| Quick Steps For Configuring Authentication Servers | 36-5 |
| Server Overview | 36-7 |
| Backup Authentication Servers | 36-7 |
| Authenticated Switch Access | 36-7 |
| Port-Based Network Access Control (802.1X) | 36-8 |
| ACE/Server | 36-9 |
| Clearing an ACE/Server Secret | 36-9 |
| RADIUS Servers | 36-10 |
| RADIUS Server Attributes | 36-10 |
| Standard Attributes | 36-10 |
| Client IP in Accounting Message | 36-12 |
| Vendor-Specific Attributes for RADIUS | 36-13 |
| Configuring Functional Privileges on the Server | 36-14 |
| RADIUS Accounting Server Attributes | 36-14 |
| Calling-Station-ID in RADIUS Access and Accounting Packets | 36-16 |
| NAS-Identifier Support in RADIUS Access Request Packets | 36-17 |
| NAS-IP Address Support in RADIUS Packets for OV Managed Switch | 36-17 |
| Configuring Case Sensitive MAC Address Authentication for RADIUS | 36-19 |
| Configuring NAS Port for RADIUS Authentication and Accounting | 36-20 |
| Configuring Unique Session ID for RADIUS Accounting | 36-22 |
| Acct-Input-Gigawords and Acct-Output-Gigawords in RADIUS Accounting Packets | 36-24 |
| Configuring the RADIUS Client | 36-25 |
| Configuring RADIUS Server Polling | 36-26 |
| Configuring RADIUS Health Check | 36-26 |
| RADIUS Server Statistics | 36-27 |
| Global Information | 36-27 |
| Authorization Statistics | 36-28 |
| Authentication Statistics | 36-28 |
| Accounting Statistics | 36-29 |
| BYOD Statistics | 36-29 |
| Viewing the RADIUS Server Statistics | 36-29 |
| Clearing the RADIUS Server Statistics | 36-30 |
| TACACS+ Server | 36-31 |
| TACACS+ Client Limitations | 36-32 |
| Configuring the TACACS+ Client | 36-32 |

| | |
|---|-------|
| LDAP Servers | 36-34 |
| Setting Up the LDAP Authentication Server | 36-34 |
| LDAP Server Details | 36-35 |
| LDIF File Structure | 36-35 |
| Common Entries | 36-35 |
| Directory Entries | 36-36 |
| Directory Searches | 36-37 |
| Retrieving Directory Search Results | 36-37 |
| Directory Modifications | 36-38 |
| Directory Compare and Sort | 36-38 |
| The LDAP URL | 36-38 |
| Password Policies and Directory Servers | 36-40 |
| Directory Server Schema for LDAP Authentication | 36-41 |
| Vendor-Specific Attributes for LDAP Servers | 36-41 |
| LDAP Accounting Attributes | 36-42 |
| Dynamic Logging | 36-44 |
| Configuring the LDAP Authentication Client | 36-45 |
| Creating an LDAP Authentication Server | 36-46 |
| Modifying an LDAP Authentication Server | 36-46 |
| Setting Up SSL for an LDAP Authentication Server | 36-46 |
| Removing an LDAP Authentication Server | 36-47 |
| Verifying the Authentication Server Configuration | 36-47 |
| Chapter 37 | |
| Configuring 802.1X | 37-1 |
| In This Chapter | 37-1 |
| 802.1X Specifications | 37-2 |
| 802.1X Defaults | 37-3 |
| Quick Steps for Configuring 802.1X | 37-4 |
| 802.1X Overview | 37-6 |
| Supplicant Classification | 37-6 |
| 802.1X Ports and DHCP | 37-7 |
| Re-authentication | 37-8 |
| Enabling 802.1x pass-through | 37-8 |
| 802.1X Accounting | 37-8 |
| Setting Up Port-Based Network Access Control | 37-9 |
| Setting 802.1X Switch Parameters | 37-9 |
| Enabling MAC Authentication | 37-9 |
| Enabling 802.1X on Ports | 37-9 |
| Configuring 802.1X Port Parameters | 37-10 |
| Configuring the Port Control Direction | 37-10 |
| Configuring the Port Authorization | 37-10 |
| Configuring 802.1X Port Timeouts | 37-10 |
| Configuring the Maximum Number of Requests | 37-11 |
| Configuring the Number of Polling Retries | 37-11 |
| Re-authenticating an 802.1X Port | 37-12 |
| Initializing an 802.1X Port | 37-12 |
| Configuring Accounting for 802.1X | 37-12 |
| Configuring 802.1x Delay Learning | 37-13 |

| | | |
|-------------------|--|-------|
| | Re-authentication Process Based on the RADIUS returned attributed Session-Timeout | 37-13 |
| | Verifying the 802.1X Port Configuration | 37-15 |
| Chapter 38 | Managing Policy Servers | 38-1 |
| | In This Chapter | 38-1 |
| | Policy Server Specifications | 38-2 |
| | Policy Server Defaults | 38-2 |
| | Policy Server Overview | 38-3 |
| | Installing the LDAP Policy Server | 38-3 |
| | Modifying Policy Servers | 38-4 |
| | Modifying LDAP Policy Server Parameters | 38-4 |
| | Disabling the Policy Server From Downloading Policies | 38-4 |
| | Modifying the Port Number | 38-5 |
| | Modifying the Policy Server Username and Password | 38-5 |
| | Modifying the Searchbase | 38-5 |
| | Configuring a Secure Socket Layer for a Policy Server | 38-6 |
| | Loading Policies From an LDAP Server | 38-6 |
| | Removing LDAP Policies From the Switch | 38-7 |
| | Interaction With CLI Policies | 38-7 |
| | Verifying the Policy Server Configuration | 38-7 |
| Chapter 39 | Configuring QoS | 39-1 |
| | In This Chapter | 39-1 |
| | QoS Specifications | 39-2 |
| | QoS General Overview | 39-3 |
| | QoS Policy Overview | 39-4 |
| | How Policies Are Used | 39-4 |
| | Valid Policies | 39-5 |
| | Policy Lists | 39-5 |
| | Interaction With Other Features | 39-6 |
| | Ethernet Service (VLAN Stacking) | 39-6 |
| | Condition Combinations | 39-7 |
| | Action Combinations | 39-9 |
| | Condition and Action Combinations | 39-11 |
| | QoS Defaults | 39-12 |
| | Global QoS Defaults | 39-12 |
| | QoS Port Defaults | 39-13 |
| | Policy Rule Defaults | 39-13 |
| | Policy Action Defaults | 39-14 |
| | Default (Built-in) Policies | 39-14 |
| | QoS Configuration Overview | 39-15 |

| | |
|---|-------|
| Configuring Global QoS Parameters | 39-15 |
| Enabling/Disabling QoS | 39-15 |
| Setting the Global Default Dispositions | 39-16 |
| Setting the Global Default Servicing Mode | 39-16 |
| Automatic QoS Prioritization | 39-16 |
| Configuring Automatic Prioritization for NMS Traffic | 39-17 |
| Configuring Automatic Prioritization for IP Phone Traffic | 39-17 |
| Using the QoS Log | 39-18 |
| What Information Is Logged | 39-18 |
| Number of Lines in the QoS Log | 39-19 |
| Log Detail Level | 39-19 |
| Forwarding Log Events | 39-19 |
| Forwarding Log Events to the Console | 39-21 |
| Displaying the QoS Log | 39-21 |
| Clearing the QoS Log | 39-22 |
| Classifying Bridged Traffic as Layer 3 | 39-22 |
| Setting the Statistics Interval | 39-22 |
| Returning the Global Configuration to Defaults | 39-23 |
| Verifying Global Settings | 39-23 |
| QoS Ports and Queues | 39-24 |
| Shared Queues | 39-24 |
| Prioritizing and Queue Mapping | 39-24 |
| Maintaining the 802.1p Priority for IP Packets | 39-25 |
| Configuring Queuing Schemes | 39-26 |
| Configuring the Servicing Mode for a Port | 39-27 |
| Bandwidth Shaping | 39-28 |
| Configuring the Egress Queue Maximum Bandwidth | 39-28 |
| Setting the DEI Bit | 39-29 |
| Configuring the DEI Bit Setting | 39-29 |
| Equal Scheduling For Yellow Traffic | 39-30 |
| To Configure Equal Scheduling of Yellow Traffic | 39-31 |
| Trusted and Untrusted Ports | 39-32 |
| QoS Profiles for Trusted and Untrusted Ports | 39-33 |
| Configuring Trusted Ports | 39-33 |
| Using Trusted Ports With Policies | 39-33 |
| Verifying the QoS Port and Queue Configuration | 39-34 |
| Creating Policies | 39-34 |
| Quick Steps for Creating Policies | 39-35 |
| ASCII-File-Only Syntax | 39-36 |
| Creating Policy Conditions | 39-37 |
| Removing Condition Parameters | 39-38 |
| Deleting Policy Conditions | 39-38 |
| Creating Policy Actions | 39-38 |
| Removing Action Parameters | 39-39 |
| Deleting a Policy Action | 39-39 |
| Creating Policy Rules | 39-39 |
| Configuring a Rule Validity Period | 39-40 |
| Disabling Rules | 39-40 |
| Rule Precedence | 39-41 |
| Saving Rules | 39-41 |
| Logging Rules | 39-42 |

| | |
|--|-------------|
| Deleting Rules | 39-42 |
| Creating Policy Lists | 39-42 |
| Guidelines for Configuring Policy Lists | 39-43 |
| Using the Default Policy List | 39-44 |
| Using Egress Policy Lists | 39-44 |
| Policy List Examples | 39-45 |
| Verifying Policy Configuration | 39-47 |
| Testing Conditions | 39-48 |
| Using Condition Groups in Policies | 39-50 |
| ACLs | 39-50 |
| Sample Group Configuration | 39-50 |
| Creating Network Groups | 39-51 |
| Creating Services | 39-52 |
| Creating Service Groups | 39-53 |
| Creating MAC Groups | 39-54 |
| Creating Port Groups | 39-55 |
| Port Group and Per Port Rate Limiting | 39-56 |
| Port Groups and Maximum Bandwidth | 39-57 |
| Creating VLAN Groups | 39-58 |
| Verifying Condition Group Configuration | 39-59 |
| Using Map Groups | 39-61 |
| Sample Map Group Configuration | 39-61 |
| How Map Groups Work | 39-62 |
| Creating Map Groups | 39-62 |
| Verifying Map Group Configuration | 39-63 |
| Applying the Configuration | 39-64 |
| Deleting the Pending Configuration | 39-65 |
| Flushing the Configuration | 39-65 |
| Interaction With LDAP Policies | 39-66 |
| Verifying the Applied Policy Configuration | 39-66 |
| Policy Applications | 39-67 |
| Basic QoS Policies | 39-68 |
| Basic Commands | 39-68 |
| Traffic Prioritization Example | 39-68 |
| Bandwidth Shaping Example | 39-69 |
| Tri-Color Marking | 39-69 |
| Configuring TCM Policies | 39-70 |
| TCM Policy Example | 39-71 |
| Redirection Policies | 39-72 |
| Policy-Based Mirroring | 39-73 |
| ICMP Policy Example | 39-74 |
| 802.1p and ToS/DSCP Marking and Mapping | 39-74 |
| Policy-Based Routing | 39-75 |
| Chapter 40 Configuring ACLs | 40-1 |
| In This Chapter | 40-1 |
| ACL Specifications | 40-2 |
| ACL Defaults | 40-3 |

| | |
|--|-------------|
| Quick Steps for Creating ACLs | 40-4 |
| ACL Overview | 40-5 |
| Rule Precedence | 40-6 |
| How Precedence is Determined | 40-6 |
| Interaction With Other Features | 40-6 |
| Valid Combinations | 40-6 |
| ACL Configuration Overview | 40-7 |
| Setting the Global Disposition | 40-7 |
| Creating Condition Groups For ACLs | 40-8 |
| Configuring ACLs | 40-8 |
| Creating Policy Conditions For ACLs | 40-9 |
| Creating Policy Actions For ACLs | 40-10 |
| Creating Policy Rules for ACLs | 40-10 |
| Layer 2 ACLs | 40-11 |
| Layer 2 ACL Example | 40-11 |
| Layer 3 ACLs | 40-12 |
| Layer 3 ACL: Example 1 | 40-12 |
| Layer 3 ACL: Example 2 | 40-13 |
| IPv6 ACLs | 40-13 |
| Multicast Filtering ACLs | 40-14 |
| Using ACL Security Features | 40-15 |
| Configuring a UserPorts Group | 40-16 |
| Configuring UserPort Traffic Types and Port Behavior | 40-16 |
| Configuring a DropServices Group | 40-17 |
| Configuring ICMP Drop Rules | 40-18 |
| Configuring TCP Connection Rules | 40-19 |
| Verifying the ACL Configuration | 40-20 |
| ACL Application Example | 40-22 |
| Chapter 41 | |
| Configuring IP Multicast Switching | 41-1 |
| In This Chapter | 41-1 |
| IPMS Specifications | 41-3 |
| IPMSv6 Specifications | 41-3 |
| IPMS Default Values | 41-4 |
| IPMSv6 Default Values | 41-5 |
| IPMS Overview | 41-6 |
| IPMS Example | 41-6 |
| Reserved IP Multicast Addresses | 41-7 |
| Configuring IPMS on a Switch | 41-8 |
| Enabling and Disabling IP Multicast Status | 41-8 |
| Enabling IP Multicast Status | 41-8 |
| Disabling IP Multicast Status | 41-8 |
| Enabling and Disabling IP Multicast Dynamic Control | 41-9 |
| Enabling IP Multicast Dynamic Control | 41-9 |

| | |
|---|-------|
| Disabling IP Multicast Dynamic Control | 41-9 |
| Enabling and Disabling IGMP Querier-forwarding | 41-10 |
| Enabling the IGMP Querier-forwarding | 41-10 |
| Disabling the IGMP Querier-forwarding | 41-10 |
| Configuring and Restoring the IGMP Version | 41-10 |
| Configuring the IGMP Version | 41-10 |
| Restoring the IGMP Version | 41-11 |
| Configuring and Removing an IGMP Static Neighbor | 41-11 |
| Configuring an IGMP Static Neighbor | 41-11 |
| Removing an IGMP Static Neighbor | 41-12 |
| Enabling and Disabling Static Neighbor Fast Convergence | 41-12 |
| Enabling the Static Neighbor Fast Convergence | 41-12 |
| Disabling the Static Neighbor Fast Convergence | 41-12 |
| Configuring and Removing an IGMP Static Querier | 41-12 |
| Configuring an IGMP Static Querier | 41-12 |
| Removing an IGMP Static Querier | 41-13 |
| Configuring and Removing an IGMP Static Group | 41-14 |
| Configuring an IGMP Static Group | 41-14 |
| Removing an IGMP Static Group | 41-14 |
| Modifying IPMS Parameters | 41-15 |
| Modifying the IGMP Query Interval | 41-15 |
| Configuring the IGMP Query Interval | 41-15 |
| Restoring the IGMP Query Interval | 41-15 |
| Modifying the IGMP Last Member Query Interval | 41-15 |
| Configuring the IGMP Last Member Query Interval | 41-16 |
| Restoring the IGMP Last Member Query Interval | 41-16 |
| Modifying the IGMP Query Response Interval | 41-16 |
| Configuring the IGMP Query Response Interval | 41-16 |
| Restoring the IGMP Query Response Interval | 41-17 |
| Modifying the IGMP Router Timeout | 41-17 |
| Configuring the IGMP Router Timeout | 41-17 |
| Restoring the IGMP Router Timeout | 41-17 |
| Modifying the Source Timeout | 41-18 |
| Configuring the Source Timeout | 41-18 |
| Restoring the Source Timeout | 41-18 |
| Enabling and Disabling IGMP Querying | 41-18 |
| Enabling the IGMP Querying | 41-19 |
| Disabling the IGMP Querying | 41-19 |
| Modifying the IGMP Robustness Variable | 41-19 |
| Configuring the IGMP Robustness variable | 41-19 |
| Restoring the IGMP Robustness Variable | 41-20 |
| Enabling and Disabling the IGMP Spoofing | 41-20 |
| Enabling the IGMP Spoofing | 41-20 |
| Disabling the IGMP Spoofing | 41-20 |
| Enabling and Disabling the IGMP Zapping | 41-21 |
| Enabling the IGMP Zapping | 41-21 |
| Disabling the IGMP Zapping | 41-21 |
| Limiting IGMP Multicast Groups | 41-21 |
| Setting the IGMP Group Limit | 41-22 |
| IPMSv6 Overview | 41-23 |
| IPMSv6 Example | 41-23 |

| | |
|---|-------|
| Reserved IPv6 Multicast Addresses | 41-23 |
| MLD Version 2 | 41-24 |
| Configuring IPMSv6 on a Switch | 41-25 |
| Enabling and Disabling IPv6 Multicast Status | 41-25 |
| Enabling IPv6 Multicast Status | 41-25 |
| Disabling IPv6 Multicast Status | 41-25 |
| Enabling and Disabling MLD Querier-forwarding | 41-26 |
| Enabling the MLD Querier-forwarding | 41-26 |
| Disabling the MLD Querier-forwarding | 41-26 |
| Configuring and Restoring the MLD Version | 41-26 |
| Configuring the MLD Version 2 | 41-26 |
| Restoring the MLD Version 1 | 41-27 |
| Configuring and Removing an MLD Static Neighbor | 41-27 |
| Configuring an MLD Static Neighbor | 41-27 |
| Removing an MLD Static Neighbor | 41-27 |
| Configuring and Removing an MLD Static Querier | 41-28 |
| Configuring an MLD Static Querier | 41-28 |
| Removing an MLD Static Querier | 41-28 |
| Configuring and Removing an MLD Static Group | 41-28 |
| Configuring an MLD Static Group | 41-28 |
| Removing an MLD Static Group | 41-29 |
| Modifying IPMSv6 Parameters | 41-30 |
| Modifying the MLD Query Interval | 41-30 |
| Configuring the MLD Query Interval | 41-30 |
| Restoring the MLD Query Interval | 41-30 |
| Modifying the MLD Last Member Query Interval | 41-30 |
| Configuring the MLD Last Member Query Interval | 41-30 |
| Restoring the MLD Last Member Query Interval | 41-31 |
| Modifying the MLD Query Response Interval | 41-31 |
| Configuring the MLD Query Response Interval | 41-31 |
| Restoring the MLD Query Response Interval | 41-31 |
| Modifying the MLD Router Timeout | 41-32 |
| Configuring the MLD Router Timeout | 41-32 |
| Restoring the MLD Router Timeout | 41-32 |
| Modifying the Source Timeout | 41-32 |
| Configuring the Source Timeout | 41-33 |
| Restoring the Source Timeout | 41-33 |
| Enabling and Disabling the MLD Querying | 41-33 |
| Enabling the MLD Querying | 41-33 |
| Disabling the MLD Querying | 41-33 |
| Modifying the MLD Robustness Variable | 41-34 |
| Configuring the MLD Robustness Variable | 41-34 |
| Restoring the MLD Robustness Variable | 41-34 |
| Enabling and Disabling the MLD Spoofing | 41-35 |
| Enabling the MLD Spoofing | 41-35 |
| Disabling the MLD Spoofing | 41-35 |
| Enabling and Disabling the MLD Zapping | 41-36 |
| Enabling the MLD Zapping | 41-36 |
| Disabling the MLD Zapping | 41-36 |
| Limiting MLD Multicast Groups | 41-36 |
| Setting the MLD Group Limit | 41-37 |

| | |
|---|-------------|
| Star-G Mode for Multicast Group | 41-38 |
| Enabling and Disabling star-G Mode Globally | 41-38 |
| Enabling and Disabling star-G Mode on a VLAN | 41-39 |
| Verifying star-G Mode Configuration | 41-39 |
| IPMS Application Example | 41-40 |
| IPMSv6 Application Example | 41-42 |
| Displaying IPMS Configurations and Statistics | 41-44 |
| Displaying IPMSv6 Configurations and Statistics | 41-45 |
| Chapter 42 | |
| Configuring IP Multicast VLAN | 42-1 |
| In This Chapter | 42-1 |
| IP Multicast VLAN Specifications | 42-2 |
| IP Multicast VLAN Defaults | 42-2 |
| IP Multicast VLAN Overview | 42-3 |
| Multicast VLAN Registration | 42-3 |
| VLAN Stacking Mode | 42-4 |
| IPMVLAN Lookup Mode | 42-4 |
| Enterprise Mode | 42-4 |
| IPMV Packet Flows | 42-5 |
| VLAN Stacking Mode | 42-5 |
| Enterprise Mode | 42-7 |
| Configuring IPMVLAN | 42-9 |
| Creating and Deleting IPMVLAN | 42-9 |
| Creating IPMVLAN | 42-9 |
| Deleting IPMVLAN | 42-9 |
| Assigning and Deleting IPv4/IPv6 Address | 42-10 |
| Assigning an IPv4/IPv6 Address to an IPMVLAN | 42-10 |
| Deleting an IPv4/IPv6 Address from an IPMVLAN | 42-10 |
| Assigning and Deleting a Customer VLAN Tag | 42-10 |
| Assigning C-Tag to an IPMVLAN | 42-10 |
| Deleting C-Tag from an IPMVLAN | 42-10 |
| Creating and Deleting a Sender Port | 42-10 |
| Creating a Sender Port in an IPMVLAN | 42-11 |
| Deleting a Sender Port from an IPMVLAN | 42-11 |
| Creating and Deleting a Receiver Port | 42-11 |
| Creating a Receiver Port in an IPMVLAN | 42-11 |
| Deleting a Receiver Port from an IPMVLAN | 42-12 |
| Associating an IPMVLAN with a Customer VLAN | 42-12 |
| IPMVLAN Application Example | 42-13 |
| Verifying the IP Multicast VLAN Configuration | 42-16 |
| Chapter 43 | |
| Diagnosing Switch Problems | 43-1 |
| In This Chapter | 43-1 |
| Port Mirroring Overview | 43-4 |
| Port Mirroring Specifications | 43-4 |

| | |
|--|-------|
| Port Mirroring Defaults | 43-4 |
| Quick Steps for Configuring Port Mirroring | 43-5 |
| Port Monitoring Overview | 43-6 |
| Port Monitoring Specifications | 43-6 |
| Port Monitoring Defaults | 43-6 |
| Quick Steps for Configuring Port Monitoring | 43-6 |
| sFlow Overview | 43-8 |
| sFlow Specifications | 43-8 |
| sFlow Defaults | 43-8 |
| Quick Steps for Configuring sFlow | 43-9 |
| Remote Monitoring (RMON) Overview | 43-11 |
| RMON Specifications | 43-11 |
| RMON Probe Defaults | 43-12 |
| Quick Steps for Enabling/Disabling RMON Probes | 43-12 |
| Switch Health Overview | 43-13 |
| Switch Health Specifications | 43-13 |
| Switch Health Defaults | 43-14 |
| Quick Steps for Configuring Switch Health Threshold Limits | 43-14 |
| Port Mirroring | 43-15 |
| What Ports Can Be Mirrored? | 43-15 |
| How Port Mirroring Works | 43-15 |
| What Happens to the Mirroring Port | 43-16 |
| Mirroring on Multiple Ports | 43-16 |
| Using Port Mirroring with External RMON Probes | 43-16 |
| Remote Port Mirroring | 43-17 |
| Creating a Mirroring Session | 43-18 |
| Unblocking Ports (Protection from Spanning Tree) | 43-19 |
| Enabling or Disabling Mirroring Status | 43-20 |
| Disabling a Mirroring Session (Disabling Mirroring Status) | 43-20 |
| Configuring Port Mirroring Direction | 43-20 |
| Enabling or Disabling a Port Mirroring Session (Shorthand) | 43-21 |
| Displaying Port Mirroring Status | 43-22 |
| Deleting A Mirroring Session | 43-22 |
| Configuring Remote Port Mirroring | 43-23 |
| Port Monitoring | 43-25 |
| Configuring a Port Monitoring Session | 43-26 |
| Enabling a Port Monitoring Session | 43-26 |
| Disabling a Port Monitoring Session | 43-26 |
| Deleting a Port Monitoring Session | 43-26 |
| Pausing a Port Monitoring Session | 43-27 |
| Configuring Port Monitoring Session Persistence | 43-27 |
| Configuring a Port Monitoring Data File | 43-27 |
| Suppressing Port Monitoring File Creation | 43-28 |
| Configuring Port Monitoring Direction | 43-28 |
| Displaying Port Monitoring Status and Data | 43-29 |
| sFlow | 43-30 |
| sFlow Manager | 43-30 |
| Receiver | 43-30 |

| | |
|---|-------|
| Sampler | 43-31 |
| Poller | 43-31 |
| Configuring a sFlow Session | 43-31 |
| Configuring a Fixed Primary Address | 43-32 |
| Displaying a sFlow Receiver | 43-32 |
| Displaying a sFlow Sampler | 43-33 |
| Displaying a sFlow Poller | 43-33 |
| Displaying a sFlow Agent | 43-33 |
| Deleting a sFlow Session | 43-34 |
| Remote Monitoring (RMON) | 43-35 |
| Ethernet Statistics | 43-36 |
| History (Control & Statistics) | 43-36 |
| Alarm | 43-36 |
| Event | 43-36 |
| Enabling or Disabling RMON Probes | 43-36 |
| Displaying RMON Tables | 43-37 |
| Displaying a List of RMON Probes | 43-37 |
| Displaying Statistics for a Particular RMON Probe | 43-38 |
| Sample Display for Ethernet Statistics Probe | 43-38 |
| Sample Display for History Probe | 43-39 |
| Sample Display for Alarm Probe | 43-39 |
| Displaying a List of RMON Events | 43-39 |
| Displaying a Specific RMON Event | 43-40 |
| Monitoring Switch Health | 43-41 |
| Configuring Resource and Temperature Thresholds | 43-42 |
| Enabling and Disabling Per-Port Health Threshold Monitoring | 43-44 |
| Displaying Health Threshold Limits | 43-45 |
| Configuring Sampling Intervals | 43-45 |
| Viewing Sampling Intervals | 43-46 |
| Viewing Health Statistics for the Switch | 43-46 |
| Viewing Health Statistics for a Specific Interface | 43-47 |
| Resetting Health Statistics for the Switch | 43-48 |

| | | |
|-------------------|--|-------------|
| Chapter 44 | Using Switch Logging | 44-1 |
| | In This Chapter | 44-1 |
| | Switch Logging Specifications | 44-2 |
| | Switch Logging Defaults | 44-3 |
| | Quick Steps for Configuring Switch Logging | 44-4 |
| | Switch Logging Overview | 44-5 |
| | Switch Logging Commands Overview | 44-6 |
| | Enabling Switch Logging | 44-6 |
| | Setting the Switch Logging Severity Level | 44-6 |
| | Specifying the Severity Level | 44-8 |
| | Removing the Severity Level | 44-9 |
| | Specifying the Switch Logging Output Device | 44-9 |
| | Enabling/Disabling Switch Logging Output to the Console | 44-9 |
| | Enabling/Disabling Switch Logging Output to Flash Memory | 44-9 |
| | Specifying an IP Address for Switch Logging Output | 44-9 |

Disabling an IP Address from Receiving Switch Logging Output 44-10
 Displaying Switch Logging Status 44-10
 Configuring the Switch Logging File Size 44-10
 Clearing the Switch Logging Files 44-11
 Displaying Switch Logging Records 44-11

Appendix A Software License and Copyright Statements

Alcatel License Agreement A-1
 ALE USA, Inc. 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. Appetitude, 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 AOS Release 6 Network Configuration Guide* describes how to set up and monitor software features that allows your switch to operate in a live network environment. The software features described in this manual are shipped standard with your OmniSwitch 6350, 6450 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 6350 Series
- OmniSwitch 6450 Series

Unsupported Platforms

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

- OmniSwitch 6250 Series
- 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 6350, 6450 benefits 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 AOS Release 6 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 helps 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, and 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 AOS Release 6 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 enables 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 6350, 6450* commands, refer to the *OmniSwitch AOS Release 6 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 (for example, 802.1Q) with which most network professionals are familiar.

Each software feature chapter includes sections that satisfies 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 helps you at each stage of the configuration process. Under each stage, we point you to the manual or manuals that is most helpful to you.

Stage 1: Gaining Familiarity with Basic Switch Functions

Pertinent Documentation: *Hardware Users Guide*
Switch Management Guide

Once you have your switch up and running, you want to begin investigating basic aspects of its hardware and software. Information about switch hardware is provided in the *Hardware Users Guide*. This guide provide 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 2: Integrating the Switch Into a Network

Pertinent Documentation: *Network Configuration Guide*

When you are ready to connect your switch to the network, you 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

User manuals can be downloaded at following

<https://businessportal2.alcatel-lucent.com>

The following are the titles and descriptions of all the related OmniSwitch 6350, 6450 user manuals:

- *OmniSwitch 6350 Hardware Users Guide*

Complete technical specifications and procedures for all OmniSwitch 6350 chassis, power supplies, and fans. Also includes comprehensive information on assembling and managing stacked configurations.

- *OmniSwitch 6450 Hardware Users Guide*

Complete technical specifications and procedures for all OmniSwitch 6450 chassis, power supplies, and fans. Also includes comprehensive information on assembling and managing stacked configurations.

- *OmniSwitch AOS Release 6 CLI Reference Guide*

Complete reference to all CLI commands supported on the OmniSwitch 6350, 6450. Includes syntax definitions, default values, examples, usage guidelines and CLI-to-MIB variable mappings.

- *OmniSwitch AOS Release 6 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 AOS Release 6 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 AOS Release 6 Transceivers Guide*

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

- *AOS Release 6.7.2 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.

- *Technical Tips, Field Notices*

Includes information published by Alcatel's Customer Support group.

Product Documentation

All products are shipped with a Product Documentation Card that provides details for downloading documentation for all OmniSwitch and other Alcatel-Lucent Enterprise data products. All user guides for the OmniSwitch Series are included on the Alcatel-Lucent Enterprise public website. This website also includes user guides for other Alcatel-Lucent Enterprise products. The latest user guides can be found on our website at:

<https://businessportal2.alcatel-lucent.com>

Technical Support

An Alcatel-Lucent Enterprise service agreement brings your company the assurance of 7x24 no-excuses technical support. You will 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-Lucent Enterprise Service Programs:

Web: <https://businessportal2.alcatel-lucent.com>

Email: ebg_global_supportcenter@al-enterprise.com.

Phone:

North America: 800-995-2696

Latin America: 877-919-9526

EMEA: +800 00200100 (Toll Free) or +1(650) 385-2193

Asia Pacific: +65 6240 8484

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 Ethernet port parameters of the switch, 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- [“Setting Ethernet Parameters for All Port Types” on page 1-7](#)
- [“Configuring Flood Rate Limiting” on page 1-9](#)
- [“Configuring Digital Diagnostic Monitoring \(DDM\)” on page 1-12](#)
- [“Configuring Energy Efficient Ethernet \(802.3az\)” on page 1-13](#)
- [“Setting Ethernet Combo Port Parameters” on page 1-18](#)
- [“Monitoring the Inter-stack Connection” on page 1-23](#)
- [“Using TDR Cable Diagnostics” on page 1-24](#)
- [“Interface Violation Recovery” on page 1-27](#)
- [“Link Fault Propagation” on page 1-31](#)
- [“Verifying Ethernet Port Configuration” on page 1-34](#)

For information about CLI commands that can be used to view Ethernet port parameters, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Ethernet Specifications

| | |
|---|---|
| IEEE Standards Supported | <p>802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD)</p> <p>802.3u (100BaseTX)</p> <p>802.3ab (1000BaseT)</p> <p>802.3z (1000Base-X)</p> <p>802.3ae (10GBase-X)</p> <p>802.3az (Energy Efficient Ethernet) is:</p> <ul style="list-style-type: none"> - only supported on copper ports and combo copper ports on OmniSwitch 6450. - LLDP feature as per 802.3az is not supported on OmniSwitch 6450. |
| Platforms Supported | OmniSwitch 6350, 6450 |
| MEF 2.0 | MEF CE 2.0 Certified |
| Ports Supported on OmniSwitch 6450 | <p>Ethernet (10 Mbps)</p> <p>Fast Ethernet (100 Mbps)</p> <p>Gigabit Ethernet (1 Gb/1000 Mbps)</p> <p>10 Gigabit Ethernet (10 Gb/10000 Mbps)</p> |
| Ports Supported on OmniSwitch 6350 | <p>Ethernet (10 Mbps)</p> <p>Fast Ethernet (100 Mbps)</p> <p>Gigabit Ethernet (1 Gb/1000 Mbps).</p> |
| License Upgrade for OmniSwitch 6450L (Base Model) | <p>Does not support Gigabit Ethernet (1Gbps) and 10 Gigabit Ethernet (10 Gbps) ports by default.</p> <p>Upon License Upgrade, OmniSwitch 6450 L supports 1 Gbps and 10 Gbps ports.</p> |
| License Upgrade for OmniSwitch 6450 | <p>Supports 1 Gbps ports by default.</p> <p>Upon License Upgrade, 6450 supports 10 Gbps ports.</p> |
| Jumbo Frame Configuration | Supported |
| Maximum Frame Size | <p>1533 Bytes (10/100 Mbps)</p> <p>9216 Bytes (1/10 Gbps)</p> |
| Maximum ports supported in a Link Fault Propagation (LFP) group. | <p>48 source ports</p> <p>48 destination ports</p> |
| Maximum number of LinkAgg supported in Link Fault Propagation (LFP) | 32 |

TDR Specifications

| | |
|---------------------|---|
| Platforms Supported | OmniSwitch 6450 (Normal & Extended TDR Test) |
| Ports Supported | Fast Ethernet (100 Mbps) Gigabit Ethernet (1 Gb/1000 Mbps) Non-Combo Copper Ports |
| TDR Test | Only one TDR test can be executed at a time. |

Ethernet Port Defaults

The following table shows Ethernet port default values:

| Parameter Description | Command | Default Value/Comments |
|-------------------------------|--|--|
| Trap Port Link Messages | trap port link | Disabled |
| Interface Configuration | flow | Up (Enabled) |
| Flood Only Rate Limiting | flow wait time | 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) 997 Mbps (10 Gigabit Ethernet) |
| Interface Alias | interfaces alias | None configured |
| Inter-Frame Gap | interfaces ifg | 12 bytes |
| Maximum Frame Size | interfaces max frame | 1553 Bytes (10/100 Mbps) 9216 Bytes (1/10 Gbps) |
| Interface Line Speed | interfaces speed | Copper Ports - Auto SFP - 1 Gbps SFP+ - 10Gbps |
| Duplex Mode | interfaces duplex | Copper Ports - Auto SFP/SFP+ - Full |
| Autonegotiation | interfaces autoneg | Copper Ports - Enabled SFP/SFP+ - Disabled |
| Crossover | interfaces crossover | Copper Ports - Auto SFP/SFP+ - MDI |
| Flow Control (pause) | interfaces pause | Disabled |

Ethernet Ports Overview

This chapter describes the Ethernet software CLI commands used for configuring and monitoring the Ethernet port parameters of the switch. 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 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 uses the SFP connector instead of the equivalent copper RJ-45 port.** However, if the SFP connector goes down, the equivalent combo port comes 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” on page 1-5](#) for more information on combo ports. In addition, refer to the *OmniSwitch AOS Release 6350/6450 Hardware Users Guide* for each type of switch.

See [“Setting Interface Line Speed for Combo Ports” on page 1-18](#) for more information on configuring combo ports.

Note: Settings for SFPs are dependent upon the type of transceiver being used. Refer to the *OmniSwitch AOS Release 6 Transceivers Guide* for information on supported SFPs.

OmniSwitch 6450 and OmniSwitch 6350 are now CE 2.0 certified across all four service definitions. CE 2.0 certification ensures service compliance to specifications and inter-working between vendors by testing product compliance across the four MEF service types — E-Line, E-LAN, E-Tree and E-Access.

Service providers with any of the Alcatel-Lucent CE 2.0 Certified products deployed can deliver CE 2.0 Certified services for any MEF service type. The CE 2.0 product certification designation applies to the tested configuration and, through compliance, to currently supported hardware and software in general.

Valid Port Settings on OmniSwitch

The following table lists the valid speed, duplex, and autonegotiation settings for the different port types:

| Port Type | User-Specified Port Speed (Mbps) Supported | User-Specified Duplex Supported | Auto Negotiation Supported? |
|-----------|--|---------------------------------|-----------------------------|
| RJ-45 | auto/10/100/1000 | auto/full/half | Yes |
| SFP/SFP+ | Dependent | Dependent | Dependent |

Note. TDR Operations is not supported on ports that use copper SFP.

See the *OmniSwitch AOS Release 6350/6450 Hardware Users Guide* for more information about the OmniSwitch hardware.

10/100/1000 Crossover Support

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 1-16](#) for more information.

Autonegotiation Guidelines

The following tables summarize the valid autonegotiation port settings between the OmniSwitch and another device.

| Remote Port | OmniSwitch Port | Supported | Note |
|-------------|-----------------|-----------|------------------------|
| Enabled | Enabled | Yes | |
| Disabled | Enabled | No | Both sides must match |
| Disabled | Disabled | Yes | |
| Enabled | Disabled | No | Both sides must match. |

General Port Settings

| Remote Port | OmniSwitch Port | Supported | Note |
|-------------|-----------------|-----------|------------------------|
| Enabled | Enabled | Yes | |
| Disabled | Enabled | Yes | Supported only for RCL |
| Disabled | Disabled | No | Not valid with RCL |
| Enabled | Disabled | No | Not valid with RCL |

Uplink Port Settings - Remote Configuration Download (RCL)

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 the switch honors or transmits and honors 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. The operational settings, as shown in the following table, override the configured settings as long as both autonegotiation and flow control are 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 1-17](#) and [“Configuring Flow Control on Combo Ports” on page 1-22](#) 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 [“Configuring Digital Diagnostic Monitoring \(DDM\)” on page 1-12](#) for information on configuring non-combo ports and see [“Setting Ethernet Combo Port Parameters” on page 1-18](#) 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 is displayed on a Network Management Station (NMS) whenever there is a change in the state of the port.

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 (for example, input, and output errors, deferred frames received, unicast packets transmitted). For information on using these commands, see the *OmniSwitch AOS Release 6 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

A traffic storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. The traffic storm control feature prevents LAN ports from being disrupted by a broadcast, multicast or unknown unicast traffic storms on a physical interfaces.

The storm control is implemented with applying the flood rate limiting for broadcast, multicast and unknown unicast storm. Earlier, single rate limit is shared between the all types of traffic. Now, additional support of storm control is implemented. In this storm control, individual rate limit can be applied on a port based on the storm type. Rate limit for each storm type can be enabled and disabled independently. Rate limit can be configured either in Mbps or PPS or in Percentage.

This storm control is applied on per port basis. Individual rate limit can be applied for each type of flood traffic (unknown unicast, multicast and broadcast) on a port.

By default, rate-limiting for multicast is disabled on a port; Broadcast and unknown-unicast rate-limiting enabled on a port. To enable the storm control, use the **interfaces flood enable** command. For example, to enable flood rate traffic on slot 4, enter:

```
-> interfaces 4 flood all enable
```

Similarly, to disable the strom control, use the command as follows:

```
-> interfaces 4 flood all disable
```

To configure the rate limit based on storm type, use the **interfaces flood rate** command. For example, to configure the rate limit on slot 4, enter:

```
-> interfaces 4 flood broadcast rate pps 500
```

Similarly, you can configure the individual rate limit for each storm type like unknown-unicast and multi-cast. For more information, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Note that, by default,

- 4Mbps rate limit shall be applied for 10 Mbps port for all storm type.
- 49Mbps rate limit shall be applied for 100 Mbps port for all storm type.
- 496Mbps rate limit shall be applied for 1G port for all storm type.
- 997Mbps rate limit shall be applied for 10G port for all storm type.

Displaying Strom Control Details

The **show interfaces flood rate** command is used to display TDR test statistics. For example:

```
-> show interfaces 4 flood rate
```

| Slot/ Port | Bcast Value | Bcast Type | Bcast Status | Ucast Value | Ucast Type | Ucast Status | Mcast Value | Mcast Type | Mcast Status |
|---------------|----------------|---------------|-----------------|----------------|---------------|-----------------|----------------|---------------|-----------------|
| 4/1 | 496 | mbps | enable | 496 | mbps | enable | 496 | mbps | disable |
| 4/2 | 49 | mbps | enable | 49 | mbps | enable | 49 | mbps | disable |
| 4/3 | 49 | mbps | enable | 49 | mbps | enable | 49 | mbps | disable |

The configured flood rate can be displayed for specific interface and also for a range of ports. For more information, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Configuring the Peak Flood Rate Value

A traffic storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. The traffic storm control feature prevents LAN ports from being disrupted by a broadcast, multicast, or unknown unicast traffic storms on a physical interfaces.

The following describes the AOS implementation of storm control:

- The **interfaces flood rate** command configures a maximum *ingress* flood rate value and minimum *ingress* low threshold value for an interface. This peak flood rate value is applied to flooded (unknown unicast - destination MAC address is unknown, broadcast - destination MAC address is FF:FF:FF:FF:FF:FF and multicast traffic - Destination MAC address is multicast address) traffic.
- The threshold types supported are - Mega bits per second, Packet per second, and % of the port speed.
- Storm control threshold cannot be accurate as its hardware dependent.
- It is possible to configure flood rate equal to the line speed. But it is recommended that you always configure the flood rate less than the line speed.
- The incoming traffic level is monitored and compared with the configured high (rate limit) and low threshold values. This comparison is per port per traffic basis. This will be an average value for a span of five seconds.
- When the incoming traffic flow on a port exceeds the configured high threshold value, the storm has to be controlled. This can be done by either rate limiting the traffic or blocking the traffic on that port. The traffic storm control continues to monitor the incoming traffic level even for the blocked/violated port. When the traffic on the violated port reaches the configured low threshold value, the port state is reset to normal state. If the low threshold is not configured, the port remains in violated state.
- 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 |
| <i>Mbps</i> (10 Gigabit Ethernet) | 997 |

Note. The default value for low threshold is '0'. This means, by default, auto recovery is not enabled. It will be enabled only by configuring low threshold.

To change the peak flood rate for broadcast traffic for an entire slot, enter **interfaces** followed by the slot number, **flood broadcast 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 broadcast rate mbps 49
```

To change the peak flood rate for all traffic types for a single port, enter **interfaces** followed by the slot number, a slash (/), the port number, **flood all 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 all rate mbps 49
```

To change the peak flood rate for all traffic types 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 all 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 all rate mbps 42
```

To change the peak flood rate and low threshold for all traffic types for an entire slot, enter **interfaces** followed by the slot number, **flood all rate**, peak flood rate value in megabits. For example, to configure the peak flood rate on slot 2 with peak flood rate and low threshold values as 49 megabits and 40 megabits, enter:

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

To specify the type of traffic eligible for rate limiting, see [“Configuring a Port Alias” on page 1-12](#) and [“Configuring a Port Alias” on page 1-12](#) for more information.

Configuring a Port Alias

The **interfaces alias** command is used to configure an alias (that is, 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 64 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 (for example, “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
```

Configuring Digital Diagnostic Monitoring (DDM)

Digital Diagnostics Monitoring allows the switch to monitor the status of a transceiver by reading the information contained on the transceiver's EEPROM. The transceiver can display Actual, Warning-Low, Warning-High, Alarm-Low and Alarm-High for the following:

- Temperature
- Supply Voltage
- Current
- Output Power
- Input Power

The transceiver is programmed with warning and alarm thresholds for predefined low and high conditions that can generate system events. If the actual value crosses the threshold value, trap can be generated.

To enable the DDM capability on the switch use the **interfaces transceiver ddm** command. For example, enter:

```
-> interfaces transceiver ddm enable
```

To enable or disable DDM trap globally for DDM warning/alarm threshold violations, use the trap keyword in the **interfaces transceiver ddm** command. For example, enter:

```
-> interfaces transceiver ddm trap enable
```

```
-> interfaces transceiver ddm trap disable
```

DDM trap can be enabled only when DDM is enabled.

Note. To take advantage of the DDM capability, the transceiver must support the DDM functionality. Not all transceivers support DDM, refer to the Transceivers Guide for additional DDM information.

Configuring Energy Efficient Ethernet (802.3az)

Energy Efficient Ethernet (EEE) is a protocol to allow ports to operate in idle or low power mode when there is no traffic to send. When EEE is enabled on a port it will advertise its EEE capability to its link partner. If the partner supports EEE they will operate in EEE mode. If the partner does not support EEE the ports will operate in legacy mode. This allows EEE capable switches to be deployed in existing networks avoiding backward compatibility issues.

- EEE is only applicable to OmniSwitch 6450 copper/combo-copper ports operating at 100/1000 Mbps speed.
- The LLDP option in IEEE 802.3az standard is not currently supported.

To enable the EEE capability on the switch use the **interfaces eee** command. For example, enter:

```
-> interfaces 1/1 eee enable
```

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” 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)
- **1000** (1000 Mbps Fast Ethernet)
- **auto** (auto-sensing, which is the default)—The auto setting automatically detects and matches the line speed of the attached device.

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

To set up a speed and duplex on a port, autonegotiation must 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.

To set up a speed and duplex on a port, autonegotiation must 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 1-16) and configure crossover settings (see [“Configuring Crossover Settings”](#) on page 1-16).

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 1-16 and [“Setting Ethernet Combo Port Parameters”](#) on page 1-18 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. 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** configures the interface or interfaces to detect crossover settings automatically. Setting crossover configuration to **mdix** configures the interface or interfaces for MDIX (Media Dependent Interface with Crossover), which is the standard for hubs and switches. Setting crossover to **mdi** configures 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 an interface honors or transmits and honors PAUSE frames. PAUSE frames are used to pause the flow of traffic temporarily between two connected devices to help prevent packet loss when traffic congestion occurs between switches.

If both autonegotiation and flow control are enabled for an interface, then autonegotiation determines how the interface processes 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 stop sending traffic temporarily 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 AOS Release 6 CLI Reference Guide*.

Setting Ethernet Combo Port Parameters

The following sections describe how to use CLI commands to configure combo ports on an OmniSwitch

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](#)” 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 does 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” 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 does 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 1-20) and configure crossover settings (see [“Configuring Crossover Settings for Combo Ports”](#) on page 1-21) 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 1-21 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 does 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. 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** configures the interface or interfaces to detect crossover settings automatically. Setting crossover configuration to **mdix** configures the interface or interfaces for MDIX (Media Dependent Interface with Crossover), which is the standard for hubs and switches. Setting crossover to **mdi** configures 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 does 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 honors or both transmits and honors PAUSE frames. PAUSE frames are used to pause the flow of traffic temporarily between two connected devices to help prevent packet loss when traffic congestion occurs between switches.

If both autonegotiation and flow control are enabled for an interface, then autonegotiation determines how the interface processes 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 AOS Release 6 CLI Reference Guide*.

Monitoring the Inter-stack Connection

The status, statistics, and counters of the stacking links (if the switch is stackable) can be monitored using the following CLI commands:

- `show stacking interfaces`: Displays the general interface information for the stacking ports.
- `show stacking interfaces status`: Displays the interface line settings (for example, speed, and mode) for the stacking ports.
- `show stacking interfaces counters`: Displays the interface counter information (for example, unicast, broadcast, and multi-cast packets received or transmitted) for the stacking ports.
- `show stacking interfaces counters errors`: Displays the interface error frame information for the stacking ports.

For more information on the inter-stack monitoring commands, see the “Ethernet Port Commands” chapter in the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Clearing the L2 Statistics for Stacking Ports

The L2 statistics for the stacking ports can be cleared. For example:

```
-> stacking interfaces 1/52 no l2 statistics
```

The L2 statistics for the slot/port 1/52 is cleared.

Using TDR Cable Diagnostics

Time Domain Reflectometry (TDR) is a feature that is used to detect cable faults. This feature is best deployed in networks where service providers and system administrators want to quickly diagnose the state of a cable during outages, before proceeding with further diagnosis.

Extended TDR is a feature that is used to know the attached cable characteristics. It is implemented by monitoring the transmitted signals amplitude received from the link partner.

Both the tests are used to find out different parameter of a cable under different scenarios.

When a TDR test is initiated, a signal is sent down a cable to determine the distance to a break or other discontinuity in the cable path. The length of time it takes for the signal to reach the break and return is used to estimate the distance to the discontinuity.

Initiating a TDR Cable Diagnostics Test

Consider the following guidelines before initiating a TDR test:

- Only one test can run at any given time, and there is no way to stop a test once it has started.
- The TDR test runs an “out-of-service” test; other data and protocol traffic on the port is interrupted when the test is active.
- TDR is supported only on copper ports and not on fiber or stacking ports.
- TDR is not supported on Link aggregate ports.
- Each time a TDR test is run, statistics from a test previously run on the same port are cleared.

A TDR test is initiated using the **interfaces tdr-test-start** CLI command. For example, the following command starts the test on port 2/1:

```
-> interfaces 2/1 tdr-test-start
```

- The TDR Test (CLIs related to TDR-Test) can run only on OmniSwitch 6450 platforms. The Extended-TDR Test (CLIs related to Extended-TDR Test) can run only on a switch with 1 GIG link capability (OS 6450).
- Extended TDR is a feature that is used to know the attached cable characteristics. It is implemented by monitoring the transmitted signals amplitude received from the link partner. Both the tests are used to find out different parameter of a cable under different scenarios. These Cable Diagnostic feature are supported and are integrated into the CPSS SDK.
- The Extended-TDR Test results (cable-length) between different pairs and between multiple readings within each pair can vary based on the link partner’s differential voltage and temperature conditions of the device under test. If the same cable is measured several times, occasionally there can be a major deviation in the reading exceeding +/- 10 meters.

A Extended-TDR test is initiated using the **interfaces tdr-extended-test-start** CLI command. For example, the following command starts the test on port 2/1:

```
-> interfaces 2/1 tdr-extended-test-start
```

- Extended TDR can work only when the gigabit link is established between the two link partners.
- Extended TDR operations cannot be performed on fiber, stacking and combo ports.

Displaying TDR Test Results

The `show interfaces tdr-statistics` command is used to display TDR test statistics. For example:

```
-> show interfaces 1/3 tdr-statistics
Legend: Pair 1 - green and white
        Pair 2 - orange and white
        Pair 3 - brown and white
        Pair 4 - blue and white
```

```
Slot/ No of Cable Pair1 Pair1  Pair2 Pair2  Pair3 Pair3  Pair4 Pair4  Test
port  pairs State State Length State Length State Length State Length Result
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
1/3   4    ok   0    ok    3    ok    3    ok    3    ok    3    success
```

The `show interfaces tdr-extended-statistics` command is used to display the results of the last Extended TDR test performed on a port. For example:

```
--> show interfaces 1/5 tdr-extended-statistics
Pair Swap
  Channel 1:straight
  Channel 2:straight
Pair Polarity
  Pair 1:positive
  Pair 2:positive
  Pair 3:positive
  Pair 4:positive
Pair Skew (in n-seconds)
  Pair 1:0
  Pair 2:0
  Pair 3:8
  Pair 4:0
Accurate Cable Length (in meters)
  Pair 1:15
  Pair 2:15
  Pair 3:15
  Pair 4:15
Downshift:No Downshift
```

The following cable states are indicated in the `show interfaces tdr-statistics command` output:

- **OK**—Wire is working properly
- **Open:**—Wire is broken
- **Short**—Pairs of wire are in contact with each other
- **Impedance Mismatch -**
 - Two cable of different quality or resistance are connected to each other through patch connector.
 - If the pair is short in a cable, it may affect the resistance of another pair hence it will result Impedance mismatch on that particular pair.
- **Unknown** - Cable diagnostic test unable to find the state of a cable.
- **Pair Swap** - Determines the channel associated with the MDI pair (cross or not for each two MDI pairs).

- **Pair Polarity**- Detects if the pairs are connected with reverse polarity (reverse on one side between two conductors in one pair)
- **Pair Skew**-The skew among the four pairs of cable (delay between pairs, in n-seconds)
- **Cable Length** - The length of the cable, in meters.
- **Downshift** - Gives the downshift status of the port, when the gigabit link cannot be established.

Clearing TDR Test Statistics

The **interfaces no tdr-statistics** command is used to clear the statistics of the last test performed on the port. There is no global statistics clear command. For example, the following command clears the TDR statistics on port 2/1:

```
-> interfaces 2/1 no tdr-statistics
```

TDR statistics from a previous test are also cleared when a new test starts on the same port.

Clearing TDR Extended Test Statistics

The **interfaces no tdr-extended-statistics** command is used to clear the statistics of the last test performed on the port. There is no global statistics clear command. For example, the following command clears the Extended TDR statistics on port 2/1:

```
-> interfaces 2/1 no tdr-extended-statistics
```

TDR extended statistics from a previous test are also cleared when a new test starts on the same port.

Interface Violation Recovery

The OmniSwitch allows features to shut down an interface when a violation occurs on that interface. To support this functionality, the following interface violation recovery mechanisms are provided:

- Manual recovery of a downed interface using the **interfaces clear-violation-all** command.
- An automatic recovery timer that indicates how much time a port remains shut down before the switch automatically brings the port back up (see “[Configuring the Violation Recovery Time](#)” on page 1-29).
- A maximum number of recovery attempts setting that specifies the number of recoveries that can occur before a port is permanently shutdown (see “[Configuring the Violation Recovery Maximum Attempts](#)” on page 1-29).
- An SNMP trap is generated each time an interface is shut down by a feature. SNMP trap can occur even when the interface is already shutdown by another feature. The trap also indicates the reason for the violation.
- An SNMP trap is generated when a port is recovered. The trap also includes information about how the port was recovered. Enabling or disabling this type of trap is allowed using the **interfaces violation-recovery-trap** command.

Violation Shutdown and Recovery Methods

A port can be shut down with one of the following methods, depending on the feature.

Filtering – The port is blocked by applying filtering to discard all packets sent or received on the port. With this method, the link LED of the port remains ON. A port in this state can be recovered using the following methods:

- Using the **interfaces clear-violation-all** command to clear the violation manually.
- Automatic recovery when the interface recovery timer expires.
- Using the **interfaces admin** command to administratively disable and enable the interface.
- Disconnecting and reconnecting the interface link.
- A link down and link up event.

Administratively – A port is administratively disabled. With this method, the LED does not remain ON. A port in this state can be recovered using only the following methods:

- Using the **interfaces clear-violation-all** command to clear the violation manually.
- Automatic recovery when the interface recovery timer expires.
- Using the **interfaces admin** command to administratively disable and enable the interface.

Disconnecting or reconnecting the interface link or a link down/up event *does not* recover a port that was administratively disabled.

Interface Violation Exceptions

An interface violation is not applied to an interface when any of the following scenarios occur:

- An interface is already in a permanent shutdown state. In this case, the only method for recovery is to use the **interface clear-violation-all** command.
- An interface is already shutdown by another feature.
- An interface is not operationally up.

Interaction With Other Features

The following table lists the features that use the interface violation recovery mechanisms, along with the violation reason and shutdown type:

| Feature | Reason Code | Shutdown Type |
|----------------------------|-------------|---------------|
| BPDU Shutdown | STP | Discard |
| User Port Shutdown | QOS | Discard |
| Policy rule - port disable | QOS | Discard |
| LPS | LPS-D | Discard |
| LPS | LPS-S | Admin-Down |
| UDLD | UDLD | Admin-Down |
| NetSec | NetSec | Admin-Down |
| NI | NISup | Admin-Down |
| LLDP Rouge Detection | LLDP | Discard |

Configuring Interface Violation Recovery

The following sections provide information about how to configure parameter values that apply to the interface violation recovery mechanisms.

Configuring the Violation Recovery Time

The violation recovery time specifies the amount of time the switch waits before automatically recovering a port that was shut down due to a violation. When the recovery timer expires, the interface is operationally re-enabled and the violation on the interface is cleared.

Consider the following when configuring the violation recover time:

- The timer value does not apply to interfaces that are in a permanent shutdown state. A port in this state is only recoverable using the **interfaces clear-violation-all** command.
- The interface violation recovery mechanism is not supported on link aggregates, but is supported on the link aggregate member ports.
- The auto recovery timer is not specific to UDLD.

The **interfaces violation-recovery-time** command is used to configure the automatic recovery time value, which is configurable on a per-port or global basis. For example, the following commands set the violation recovery time to 600 secs at the global level and to 200 secs for port 2/1:

```
-> interfaces violation-recovery-time 600
-> interfaces 2/1 violation-recovery-time 200
```

The violation recovery time value configured for a specific interface overrides the global value configured for all switch interfaces. To set the port-level value back to the global value, use the **default** parameter with the **interfaces violation-recovery-time** command. For example, the following command sets the violation recovery time for port 2/1 back to the global value of 600:

```
-> interfaces 2/1 violation-recovery-time default
```

To disable the violation recovery timer mechanism, set the recovery time to zero. For example:

```
-> interfaces violation-recovery-time 0
-> interfaces 2/1 violation-recovery-time 0
```

Configuring the Violation Recovery Maximum Attempts

The violation recovery maximum setting specifies the maximum number of recovery attempts allowed before a port is permanently shut down. This value increments by one whenever an interface recovers from a violation using the automatic recovery timer mechanism. When the number of recovery attempts exceeds this configured threshold, the interface is permanently shut down. The only way to recover a permanently shutdown interface is to use the **interfaces clear-violation-all** command.

The recovery mechanism tracks the number of recoveries within a fixed time window (FTW). The $FTW = 2 * \text{maximum recovery number} * \text{recovery timer}$. For example, if the maximum number of recovery attempts is set to 4 and the recovery timer is set to 5, the FTW is 40 secs ($2 * 4 * 5=40$).

The **interfaces violation-recovery-maximum** command is used to configure the maximum number of recovery attempts. This value is configurable on a per-port or global basis. For example, the following commands set the number of attempts to 3 at the global level and to 5 for port 2/1:

```
-> interfaces violation-recovery-maximum 3
-> interfaces 2/1 violation-recovery-maximum 5
```

The maximum recovery attempts value configured for a specific interface overrides the global value configured for all switch interfaces. To set the port-level value back to the global value, use the **default** parameter with the **interfaces violation-recovery-maximum** command. For example, the following command sets the number of recovery attempts for port 2/1 back to the global value of 3:

```
-> interfaces 2/1 violation-recovery-maximum default
```

To disable the violation recovery maximum attempts mechanism, set the number of attempts to zero. For example:

```
-> interfaces violation-recovery-maximum 0  
-> interfaces 2/1 violation-recovery-maximum 0
```

Verifying the Interface Violation Recovery Configuration

Use the following **show** commands to verify the violation recovery configuration:

| | |
|---|--|
| show interfaces port | Displays the administrative status (up or down), link status, violations, recovery time, maximum recovery attempts. |
| show interfaces violation-recovery | Displays the globally configured recovery time, SNMP recovery trap enable or disable status and maximum recovery attempts. |

For more information about the resulting displays from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Link Fault Propagation

The Link Fault Propagation (LFP) feature provides a mechanism to propagate a local interface failure into another local interface. In many scenarios, a set of ports provide connectivity to the network. If all these ports go down, the connectivity to the network is lost. However, the remote end remains unaware of this loss of connectivity and continues to send traffic that is unable to reach the network. To solve this problem, LFP does the following:

- Monitors a group of interfaces (configured as source ports).
- If all the source ports in the group go down, LFP waits a configured amount of time then shuts down another set of interfaces (configured as destination ports) that are associated with the same group.
- When any one of the source ports comes back up, all of the destination ports are brought back up and network connectivity is restored.

The LFP source and destination ports can be physical or link aggregation ports. If the destination port is a link aggregation port the shutdown consists of shutting down all members of the link aggregation group (physically down). However, the link aggregation group remains administratively enabled.

Interaction With Interfaces Violation Recovery

- The `interfaces clear-violation-all` command will clear the LFP violations and mark the interfaces as up even if the violation condition still exists.
- An admin down followed by an admin up will clear the LFP violation and mark the interfaces as up even if the violation condition still exists.
- When the destination port is a link aggregate, the shutdown action does not shutdown the link aggregation. Instead, all the ports that are members of the link aggregation at the time of the violation are shutdown.
- A link aggregate port remains in a violation state even if the port leaves the link aggregate.
- If a port that is not a member of a link aggregate at the time a violation occurred is added to a link aggregate, the switch will not shut down the port.
- SNMP traps cannot be configured for LFP. The interface violation recovery mechanism will be responsible for sending traps when a port is shutdown or recovered by LFP.
- If the wait-to-restore (WTR) timer is configured on the source ports of a LFP group with link monitoring enabled, the state of the destination ports of the group will be determined by the link state of the ports after the WTR timer has expired.

See [“Interface Violation Recovery”](#) on page 1-27 for more information.

Configuring Link Fault Propagation

Configuring LFP requires the following steps:

1 Create an LFP group. This type of group identifies the source ports to monitor and the destination ports to bring down when all of the source ports go down. To create an LFP group, use the [link-fault-propagation group](#) command. For example:

```
-> link-fault-propagation group 1
```

2 Associate source ports with the LFP group. To associate source ports to an LFP group, use the [link-fault-propagation group source](#) command. For example:

```
-> link-fault-propagation group 1 source port 1/2-5 2/3
```

3 Associate destination ports with the LFP group. To associate destination ports with an LFP group, use the [link-fault-propagation group destination](#) command. For example:

```
-> link-fault-propagation group 1 destination port 1/5-8 2/3
```

4 Configure the LFP wait-to-shutdown timer. This timer specifies the amount of time that LFP will wait before shutting down all the destination ports. To configure this timer value, use the [link-fault-propagation group wait-to-shutdown](#) command. For example:

```
-> link-fault-propagation group 1 wait-to-shutdown 70
```

5 Enable Admin State of the LFP group. To administratively enable Link Fault Propagation on a group, use the [link-fault-propagation group admin-status](#) command. For example:

```
-> link-fault-propagation group 1 admin-status enable
```

Note. *Optional.* To verify the LFP configuration, use the [show link-fault-propagation group](#) command. For example:

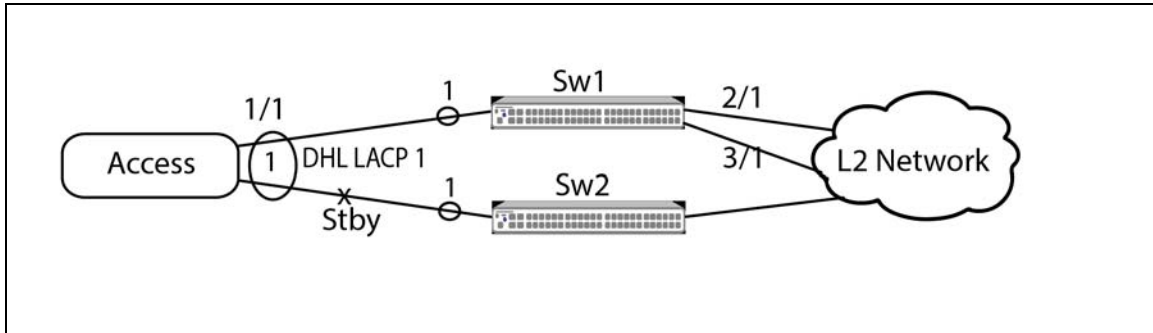
```
-> show link-fault-propagation group
Group Id : 2
Source Port(s)      : 0/1-2 1/1-5 1/7,
Destination Port(s) : 0/3 1/10-13,
Group-Src-Ports Status : up,
Admin Status        : enable,
Wait To Shutdown    : 10
```

```
Group Id : 6
Source Port(s)      : 1/2 1/6 1/9,
Destination Port(s) : 1/10-11 1/13,
Group-Src-Ports Status : down,
Admin Status        : disable,
Wait To Shutdown    : 5
```

See the *OmniSwitch AOS Release 6 CLI Reference Guide* for more information about LFP commands.

LFP Application Example: Dual-Home Link

This section provides an example of using LFP in a Dual-Home Link (DHL) configuration, as shown in the following sample DHL topology:



Link Fault Propagation - Application Example

In this DHL example:

- When interfaces 2/1 and 3/1 on Sw1 are down, the access switch will keep interface 1/1 as active and traffic will still be forwarded to Sw1 even though it has no network connectivity.
- To allow DHL to switch to the standby interface, LACP 1 on Sw1 would need to be disabled so that interface 1/1 on the access switch leaves the LACP group.

```
-> link-fault-propagation group 1
-> link-fault-propagation group 1 source port 2/1 3/1 destination linkagg 1
-> link-fault-propagation group 1 wait-to-shutdown 40
-> link-fault-propagation group 1 admin-status enable
```

For more information, see [“Configuring Dual-Home Links” on page 11-1](#)

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 (for example, packets received or transmitted, and deferred frames received). |
| show interfaces counters | Displays interface counters information (for example, unicast, broadcast, and multi-cast packets received or transmitted). |
| show interfaces counters errors | Displays interface error frame information (for example, CRC errors, transit errors, and receive errors) |
| show interfaces collisions | Displays interface collision information (for example, number of collisions and number of retries) |
| show interfaces status | Displays interface line settings (for example, speed, and mode). |
| show interfaces port | Displays the administrative status (up or down), link status, violations, recovery time, maximum recovery attempts, along with the reason for violation in case link status of port is down and the value of the wait-to-restore timer for the specified port or ports. |
| show interfaces ifg | Displays interface inter-frame gap values. |
| show interfaces flood rate | Displays configured flood rate settings. |
| show interfaces traffic | Displays interface traffic statistics. |
| show interfaces capability | Displays default autonegotiation, speed, duplex, flow, and cross-over settings for a single port, a range of ports, or all ports on a Network Interface (NI) module. |
| show interfaces hybrid | Displays general interface information (for example, 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 for combo ports. |
| show interfaces hybrid pause | Displays the flow control pause configuration for combo ports. |
| show interfaces hybrid capability | Displays default autonegotiation, speed, duplex, flow, and cross-over settings for a single combo port, a range of combo ports, or all combo ports on a switch. |
| show interfaces hybrid accounting | Displays interface accounting information (for example, packets received/transmitted, deferred frames received) for combo ports. |
| show interfaces hybrid counters | Displays interface counters information (for example, unicast, broadcast, multi-cast packets received or transmitted) for combo ports. |
| show interfaces hybrid counters errors | Displays interface error frame information (for example, CRC errors, transit errors, receive errors) for combo ports. |
| show interfaces hybrid collisions | Displays interface collision information (for example, 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. |
| show link-fault-propagation group | Displays details of a Link Fault Propagation group. |

These commands can be 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 AOS Release 6 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 AOS Release 6 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.](#)
- [“Configuring Hash Chain Length” on page 2-11](#)
- [“Displaying Source Learning Information” on page 2-12.](#)

Source Learning Specifications

The functionality described in this chapter is supported on the OmniSwitch 6350, 6450 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 6350, 6450 = 16K/stack |
| Maximum number of static L2 multicast MAC addresses. | OmniSwitch 6350, 6450 = 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 | enabled |
| Hash control chain length | hash-control chain-length | default (Hash Chain Length will be set to 4) |

Sample MAC Address Table Configuration

The following steps provide a quick tutorial that creates 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 39, "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 AOS Release 6 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 25, “Configuring Static Link Aggregation”](#) and [Chapter 26, “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 receives 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, and so on, 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 AOS Release 6 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 includes 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 assigns the multicast-VLAN association created with the **mac-address-table static-multicast** to multicast group one.

Note. 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 (for example, **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 has 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 28, "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 AOS Release 6 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.

Configuring Hash Chain Length

Changing the hash-mode to CRC is a more efficient technique than XOR mode. Since the MAC collision is hardware and algorithm dependent, it may further reduce the probability of MAC collision by using the provision in hardware to increase the bucket size from current set value of 4 to a higher value of 8.

To configure the hash chain length in the hardware, use the **hash-control chain-length** command:

```
-> hash-control chain-length default
-> hash-control chain-length extend
```

Depending upon this configuration, the hashing bucket size for the hardware table will be decided. For more information about this command, see *OmniSwitch AOS Release 6 CLI Reference Guide*

After using this command, save the configurations using **write memory** command and reload the switch to reflect the hash length changes in the switch.

Note. Without performing a reboot (after change in hash length), actions like inserting a new NI or doing takeover should not be done.

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. |
| show hash-control chain-length | Displays the configured value for the depth of the hashing bucket. |

For more information about the resulting displays from these commands, see the *OmniSwitch AOS Release 6 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.

Learned Port Security (LPS) provides a mechanism for authorizing source learning of MAC addresses on Ethernet and Gigabit Ethernet ports. LPS does not support 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling LPS for a port on [page 3-9](#).
- Specifying a source learning time limit for all LPS ports on [page 3-10](#).
- Configuring the maximum number of MAC addresses learned per port on [page 3-16](#).
- Configuring the maximum number of filtered MAC addresses learned per port on [page 3-17](#).
- Configuring a list of authorized MAC addresses for an LPS port on [page 3-17](#).
- Configuring a range of authorized MAC addresses for an LPS port on [page 3-18](#).
- Selecting the security violation mode for an LPS port on [page 3-20](#).
- Displaying LPS configuration information on [page 3-21](#).

For more information about source MAC address learning, see [Chapter 3, “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 6350, 6450 |
| 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 | 1000 |
| Maximum number of configurable MAC address ranges per LPS port | 8 |
| Maximum number of learned MAC addresses per switch | 16K |
| Maximum number of configured MACs when MAC-move is enabled | 64 |
| Maximum bridging MACs when MAC-move is enabled | |

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

LPS is supported on Ethernet and Gigabit Ethernet fixed, mobile, tagged and authenticated ports. LPS is not supported on link aggregate and tagged (trunked) link aggregate ports.

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 admin-status 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
```

Optional: Provide infinite learning window mode where the learning window does not expire. Infinite learning window can be configured for all the LPS learning options when the shutdown value is set to zero:

```
-> port-security shutdown 0
```

Optional: The MAC addresses learned during the learning window are directly converted to static with learn-as-static option enabled, per port or globally when no-aging is enabled.

```
-> port-security shutdown 30 no-aging enable learn-as-static enable
```

Note. See [“Configuring Automatic Conversion of MAC Addresses”](#) on page 3-15, [“Configuring MAC Movement”](#) on page 3-15 , and [“Configuring Infinite Learning Window”](#) on page 3-14 for configuration options on **port-security shutdown** command.

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 [show port-security](#) command. For example:

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

```
Legend: Mac Address: * = Duplicate Static
        Mac Address: # = Pseudo Static
Port: 1/2
Operation Mode      :          ENABLED,
Max MAC bridged    :          6,
Trap Threshold     :          DISABLED,
Max MAC filtered   :          5,
Low MAC Range      :          00:00:00:00:00:00,
High MAC Range     :          ff:ff:ff:ff:ff:ff,
Violation          :          RESTRICT,
Violating MAC      :          NULL
```

```
MAC Address      VLAN  TYPE
-----+-----+-----
00:00:00:00:00:01    1  STATIC
00:00:00:00:00:02    1  STATIC(*)
00:00:00:00:00:02    1  STATIC(#)
00:00:00:00:00:13    1  STATIC
00:00:00:00:00:14    1  STATIC
00:00:00:00:00:20    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      = 25 min,
Convert-to-static        = DISABLED,
No Aging                  = ENABLED,
Boot Up                   = ENABLED,
Learn As Static          = DISABLED,
Mac Move                  = DISABLED,
Remaining Learning Window = 882 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 the number of MAC addresses learned on the port below the maximum Filtered MAC 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 | No LPS violation; MAC learned as filtered |
| 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 the learning window expires the system will learn the filtering MACs up to the maximum limit and the LPS port will go on violation.

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 3-20). 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 learned 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 the switch configuration file is saved, all dynamically learned MAC addresses in the LPS table are cleared.

Note. A dynamic MAC address learned on an LPS port is flushed when a port goes down, or MAC ages out, or the MAC address entry in the LPS table is not saved.

On enabling "no-aging" on an LPS port, the MAC addresses are automatically learned as pseudo static MAC addresses during the LPS learning window time period. These learned MAC addresses are not affected by aging and flushing operations that occur during the learning window.

Once the learning window expires, if the 'convert-to-static' option is disabled, these MAC addresses remain as pseudo static. Else if the 'convert-to-static' is enabled, the pseudo static MAC addresses are converted to static address.

Static Configuration of Authorized MAC Addresses

Authorized source MAC address entries can be configured into the LPS table as static addresses. This type of entry is similar to dynamically configured entries that authorize port access to traffic with a matching source MAC address.

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

Source learning of configured authorized MAC addresses is 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 3-17](#) and [“Configuring an Authorized MAC Address Range” on page 3-18](#) 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.

Static MAC Address Movement

You can configure same static LPS MAC on multiple LPS ports. A static LPS MAC is allowed to move between ports belonging to the same VLAN. The system supports a maximum of 64 such entries.

Example:

```
-> vlan 2
-> vlan 2 port default 1/3
-> vlan 2 port default 1/4
-> port-security 1/3 mac 00:00:00:00:00:01
-> port-security 1/4 mac 00:00:00:00:00:01
```

Note.

- Static MAC Address movement is not allowed on LPS ports configured as UNI ports.
 - System supports static MAC moves only on the LPS ports where static MAC is configured on different ports in a given VLAN.
 - When static MAC is configured on different LPS ports in a VLAN, the static MAC is valid only on one port. This port is either an ingress port or the first port on which LPS static MAC is configured.
-

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 3, “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, shutdown or discard.
- 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.

If the LPS port is shut down or the network device is disconnected from the port, the LPS table entries and the source learning MAC address table entries for the 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 override this behavior, a dynamic MAC address can be converted to a static MAC address using the [port-security convert-to-static](#) command.

To view the contents of the LPS table, use the [show port-security](#) command. Refer to the *OmniSwitch AOS Release 6 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 3-9](#).
- 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 3-10](#).
- 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 3-16](#).
- 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 3-17](#)
- Configuring one or more static authorized MAC addresses. This procedure is described in [“Configuring Authorized MAC Addresses” on page 3-17](#).
- 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 3-20](#).

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 admin-status enable
```

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

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

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 admin-status disable
```

To convert all learned bridge MAC address on LPS port into static MAC address, use the **port-security chassis** command with the **convert-to-static** parameter. For example:

```
-> port-security chassis convert-to-static
```

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.

To disable source learning on the specified LPS port(s), use the **port-security** command with the **locked** parameter. For example, the following command disables source learning on port 3 of slot 4:

```
-> port-security 4/3 admin-status locked
```

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 aging out, 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
```

When the **no-aging** parameter is enabled with the **port-security shutdown** command, all the bridged LPS MAC addresses learned during the learning window are not aged-out from the system. These MAC addresses are learned as pseudo static MAC addresses. For example:

```
-> port-security shutdown 60 no-aging enable
```

The bridged LPS MACs will be removed from the system when the **no port-security** command or **no mac-address-table** command is issued.

To start the learning window automatically at boot-up time or on switch restart, use the **port-security shutdown** command with the **boot-up** parameter enabled. For example:

```
-> port-security shutdown 60 boot-up enable
```

Note.

- The number of converted static MAC addresses cannot exceed the maximum number of MAC addresses allowed on the LPS ports.
- The conversion of dynamic MAC addresses to static ones does not apply to LPS mobile and authenticated ports.

Configuring MAC Movement for Pseudo Static MAC

A pseudo static MAC is allowed to move from one port to another, with 'mac-move' option enabled. Unlike duplicate static mac, no information will be retained on the old port upon pseudo-static mac movement. This can be used only when '**no-aging**' option is enabled. To enable MAC movement for pseudo static MAC, use the **port-security shutdown** command with the **mac-move** parameter. For example, to enable MAC movement for 20 minutes across all LPS ports within the same VLAN, enter:

```
-> port-security shutdown 20 no-aging enable mac-move enable
```

Learning Window Behavior

The following table displays the behavior of the learning window based on the combination of the learning options:

| port-security shutdown num | mac-move disable | mac-move enable | mac-move disable | mac-move enable |
|---|--|---|--|---|
| | learn-as-static disable | learn-as-static disable | learn-as-static enable | learn-as-static enable |
| no-aging enable convert-to-static enable | The MAC is learned as pseudo static and is not subject to MAC aging. The dynamically learned MAC addresses is converted to static MAC. | The MAC is learned as pseudo static and is not subject to MAC aging. For a duplicate MAC (during the learning window) MAC movement will be allowed. The dynamically learned MAC addresses is converted to static MAC. | The MAC is learned as pseudo static and is not subject to MAC aging. The MAC is directly learned as static MAC during the learning window. | The MAC is learned as pseudo static and is not subject to MAC aging. The MAC is directly learned as static MAC during the learning window. For a duplicate MAC (during the learning window) MAC movement will be allowed. |
| no-aging disable convert-to-static disable | The MAC is learned as pseudo static and is subject to MAC aging. | Option not supported in this mode. no-aging option must be enabled to allow MAC movement. | Option not supported in this mode. no-aging option must be enabled to allow direct learning of MAC to static MAC. | Option not supported in this mode. no-aging option must be enabled to allow MAC movement and direct learning of MAC to static MAC. |

The following table displays the behavior of switch and ports and when learning window is active and after expiry of learning window.

| port-security shutdown num | Behavior during learning window | Behavior after expiry of learning window. |
|---|--|--|
| no-aging enable convert-to-static enable learn-as-static enable mac-move disable | 1. Learn the MAC as static 2. update the boot.cfg file with the new static MAC. At port level MAC is learned as static. | no action |

| port-security shutdown <i>num</i> | Behavior during learning window | Behavior after expiry of learning window. |
|--|---|--|
| no-aging enable convert-to-static enable learn-as-static disable mac-move enable | 1.For a duplicate MAC learned during the learning window, mac-movement is allowed 2.MAC is learned as static on new port. 3.No information is maintained regarding the old port. | Since convert-to-static is enabled, pseudo-static MACs are converted to static. |
| no-aging enable convert to static enable learn-as-static enable mac-move enable | 1. A new MAC is learned as static 2.For a duplicate static MAC mac-movement is allowed. 3.As the MAC is already present on old port as permanent static, the entry is not deleted, but marked as duplicate static (*)and MAC on the new port is learned as pseudo-static (#). | MAC learned as permanent static. no action |
| no-aging enable convert to static disable learn-as-static enable mac-move disable | MAC learned as static in boot.cfg file and at port level with the new static MAC. | no action |
| no-aging enable convert to static disable learn-as-static disable mac-move enable | 1.For a duplicate MAC learned during the learning window, mac-movement is allowed 2.MAC is learned as static on new port. 3.No information is maintained regarding the old port. | The MAC learned as pseudo-static is not converted to static. |

For example, when:

```
-> no-aging enable convert-to-static enable mac-move disable learn-as-static
disable
```

```
-> no-aging enable convert-to-static enable mac-move enable learn-as-static
disable
```

The dynamically learned (pseudo-static) MAC addresses automatically convert-to-static after learning window expires.

Configuring Infinite Learning Window

In infinite learning window mode the learning window will not expire. Infinite learning window can be configured for all the LPS learning options by setting the shutdown value to zero. Use the **port-security shutdown** command to configure the infinite learning window. For example, to configure the infinite learning window for no-aging, convert-to-static, and boot-up, enter:

```
-> port-security shutdown 0 no-aging enable convert-to-static enable boot-up enable
```

Note. The **port-security shutdown 0** default option can be used to set all the options for learning window to their default values. For example:

```
-> port-security shutdown 0 default
```

Note. Infinite Learning Window

Infinite learning window has same behavior as learning window, but here the **convert-to-static** option is not valid. Hence when an infinite learning window is enabled, **convert-to-static** option is disabled automatically.

Configuring Automatic Conversion of MAC Addresses

The MAC addresses learned during the learning window are directly converted to static even if the **convert-to-static** option is not enabled.

When '**learn-as-static**' option is enabled, MACs are directly learned as static during learning window even if **convert-to-static** option is not enabled per port or globally when learning window is active.

This can be used only when '**no-aging**' option is enabled. To directly convert the MAC addresses to static, use the **port-security shutdown** command with the **learn-as-static** parameter. For example, to perform source learning for 20 minutes across all LPS ports and to convert the learned dynamic MAC addresses directly to static MAC addresses, enter:

```
-> port-security shutdown 20 no-aging enable learn-as-static enable
```

Configuring MAC Movement

When **mac-move** is enabled, pure static MACs are learned as static on new port and marked as duplicate MAC entries on old port. Thus duplicate MAC entries are stored on multiple ports.

MAC movement behavior for configured static MACs is as follows:

- When a MAC is learned as static, the MAC address is stored when it comes to any port other than the origin port, in any slot in the switch. This entry is stored on all the slots.
- A port specific with forwarding" action is applied. When **mac-move** is enabled, MACs are stored on all the ports except the one on which the MAC has come originally.
- Mac-move can not be enabled if total number of configured MACs are greater than 64 or when total maximum bridging count at system level is greater than or equal to 64.
- When **mac-move** is disabled then, a port specific entry with action is created in the system for all duplicate static MACs at that instance.
- When **mac-move** is disabled, the 64 MAC restriction does not apply.

If a pseudo static MAC learned is present on more than one port in the same VLAN, the MAC is allowed to move to the new port and is learned as pseudo-static MAC on the new port.

The option '**mac-move**' in learning window, allows pseudo-static MACs to move from one port to another based on condition applied. This can be used only when '**no-aging**' option is enabled.

To enable MAC movement for pseudo-static MAC, use the **port-security shutdown** command with the **mac-move** parameter. For example, to enable MAC movement for 20 minutes across all LPS ports within the same VLAN, enter:

```
-> port-security shutdown 20 no-aging enable mac-move enable
```

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 1000. 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
```

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 five 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

The MAC addresses entering the LPS enabled port is learnt as filtered MAC when the learning window expires, or the maximum MAC addresses allowed limit is reached, or the MAC is not in the allowed range of the MAC addresses for the port. The maximum number of filtered MAC addresses that can be learned is limited by a configurable parameter "max-filtering". This functionality provides logging of the MAC addresses that attempted to enter the LPS enabled port after the expiry of the learning window.

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 0 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 3-20](#).

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
```

Multiple MAC range can be configured for a port. The switch supports up to eight MAC range per port. The following configuration is a sample example of multiple MAC range configured for a port:

```
-> port-security 1/5 mac-range low 00:01:01:22:22:56 high 00:01:01:22:22:67
-> port-security 1/5 mac-range low 00:01:01:22:33:56 high 00:01:01:22:33:67
-> port-security 1/5 mac-range low 00:01:01:22:44:56 high 00:01:01:22:44:67
-> port-security 1/5 mac-range low 00:01:22:22:11:56 high 00:01:22:22:11:67
-> port-security 1/5 mac-range low 00:01:22:22:22:56 high 00:01:22:22:22:67
-> port-security 1/5 mac-range low 00:01:22:22:33:56 high 00:01:22:22:33:67
-> port-security 1/5 mac-range low 00:01:22:22:44:56 high 00:01:22:22:44:67
-> port-security 1/5 mac-range low 00:01:22:22:55:56 high 00:01:22:22:55:67
```

Note. When a new MAC range is configured, the default MAC range is replaced by the configured MAC range. The default MAC range is automatically applied when all the configured MAC range for the port is deleted.

To modify or delete the configured MAC range use the **no** form of the command. To modify the configured MAC range, the existing configuration must be deleted and the new configuration can be added. For example, to delete the MAC range low 00:01:01:22:44:56 high 00:01:01:22:44:67 configured for the port 1/5 in the above example:

```
-> no port-security 1/5 mac-range low 00:01:01:22:44:56 high 00:01:01:22:44:67
```

To view the configured MAC range for the port, use the **show port-security** command. For example:

```
-> show port-security mac-range

Port                Low MAC                High MAC
-----+-----+-----
```

```

1/5  00:01:01:22:22:56 00:01:01:22:22:67
1/5  00:01:01:22:33:56 00:01:01:22:33:67
1/5  00:01:01:22:44:56 00:01:01:22:44:67
1/5  00:01:22:22:11:56 00:01:22:22:11:67
1/5  00:01:22:22:22:56 00:01:22:22:22:67
1/5  00:01:22:22:33:56 00:01:22:22:33:67
1/5  00:01:22:22:44:56 00:01:22:22:44:67
1/5  00:01:22:22:55:56 00:01:22:22:55:67
1/3  00:00:00:00:00:38 00:00:00:00:00:40
1/3  00:00:00:00:00:41 00:00:00:00:00:44
1/3  00:00:00:00:00:45 00:00:00:00:00:48
1/3  00:00:00:00:00:49 00:00:00:00:00:52

```

```
-> show port-security 1/5 mac-range
```

| Port | Low MAC | High MAC |
|------|-------------------|-------------------|
| 1/5 | 00:01:01:22:22:56 | 00:01:01:22:22:67 |
| 1/5 | 00:01:01:22:33:56 | 00:01:01:22:33:67 |
| 1/5 | 00:01:01:22:44:56 | 00:01:01:22:44:67 |
| 1/5 | 00:01:22:22:11:56 | 00:01:22:22:11:67 |
| 1/5 | 00:01:22:22:22:56 | 00:01:22:22:22:67 |
| 1/5 | 00:01:22:22:33:56 | 00:01:22:22:33:67 |
| 1/5 | 00:01:22:22:44:56 | 00:01:22:22:44:67 |
| 1/5 | 00:01:22:22:55:56 | 00:01:22:22:55:67 |

Specify the slot and port to view the MAC range configured for the specific port else the MAC range configured for all the ports is displayed.

Refer to the *OmniSwitch AOS Release 6 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 the restrict mode, the traffic for the MAC addresses learned prior to violation are allowed. The learned MAC addresses are retained. No other learning is allowed on the port.

In the shutdown mode, the physical link is brought down and no traffic is allowed on the port. All dynamically learned MAC addresses are removed. After a shutdown occurs, a manual reset is required to return the port back to normal operation. No traffic is allowed in this violation mode.

In the discard mode, the physical link is up. The port is in discard state and no traffic is allowed on the port. All dynamically learned MAC addresses are removed. No traffic is allowed in this violation mode.

When a port is shut down or goes into discard mode, disable and enable LPS on that port or use the `port-security release` command to restore the port to normal operation. When a port goes into restrict mode, use the **port-security release** command to restore the port 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** or **discard**. 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  
-> port-security 3/4 violation discard
```

Displaying Learned Port Security Information

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

| | |
|------------------------------------|--|
| show port-security | 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. |
| show port-security brief | Displays the per port LPS parameters configured for all the ports. |

For more information about the resulting display from these commands, see the *OmniSwitch AOS Release 6 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 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- “[Creating/Modifying VLANs](#)” on page 4-5.
- “[Defining VLAN Port Assignments](#)” on page 4-6.
- “[Enabling/Disabling VLAN Mobile Tag Classification](#)” on page 4-8.
- “[Enabling/Disabling Spanning Tree for a VLAN](#)” on page 4-9.
- “[Configuring VLAN Router Interfaces](#)” on page 4-10.
- “[Bridging VLANs Across Multiple Switches](#)” on page 4-10.
- “[Verifying the VLAN Configuration](#)” on page 4-12.

For information about statically and dynamically assigning switch ports to VLANs, see [Chapter 7](#), “[Assigning Ports to VLANs](#).”

For information about defining VLAN rules that allow dynamic assignment of mobile ports to a VLAN, see [Chapter 9](#), “[Defining VLAN Rules](#).”

For information about Spanning Tree, see [Chapter 12](#), “[Configuring Spanning Tree](#).”

For information about routing, see [Chapter 28](#), “[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 6350, 6450 |
| Maximum VLANs per switch | 4094 |
| Maximum VLAN port associations (VPA) per switch | 32768 |
| Maximum 802.1Q VLAN port associations per switch | 2500 |
| 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 AOS Release 6 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 creates 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
          stree          mble
vlan  type admin oper  1x1   flat   auth   ip   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 (for example, 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 enables 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 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 (for example, 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 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 receives 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 4-6](#) and [Chapter 7, “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 4-10](#).

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 4-9](#) 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 4-12](#) 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 9, “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 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 24, “Configuring 802.1Q.”](#))
- Configuring ports as members of a link aggregate that is assigned to a configured default VLAN. (See [Chapter 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 7, “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 4-8.)
- Packet contents matches criteria defined in a VLAN rule. (See [“Configuring VLAN Rule Classification”](#) on page 4-8 and [Chapter 9, “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 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 participates in dynamic VLAN assignment. See [Chapter 7, “Assigning Ports to VLANs,”](#) for detailed procedures.
- 2** Enable/disable mobile port properties that determine mobile port behavior. See [Chapter 7, “Assigning Ports to VLANs,”](#) for detailed procedures.
- 3** Create VLANs that receives 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 triggers 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 4-8](#).

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 has to 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 7, “Assigning Ports to VLANs,”](#) and [Chapter 9, “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 9, “Defining VLAN Rules.”](#)

| Rule Types | Command |
|-----------------|---|
| DHCP | <code>vlan dhcp mac</code> <code>vlan dhcp mac range</code> <code>vlan dhcp port</code> <code>vlan dhcp generic</code> |
| MAC address | <code>vlan mac</code> <code>vlan mac range</code> |
| Network address | <code>vlan ip</code> <code>vlan protocol</code> |
| Protocol | <code>vlan protocol</code> |
| Port | <code>vlan port</code> |

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 receives 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 24, “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 excludes 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 includes 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 AOS Release 6 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 12, “Configuring Spanning Tree.”](#)

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 (for example, IP - 21.0.0.10).

Alcatel 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 28, “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 4-10](#) 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 AOS Release 6 CLI Reference Guide*.

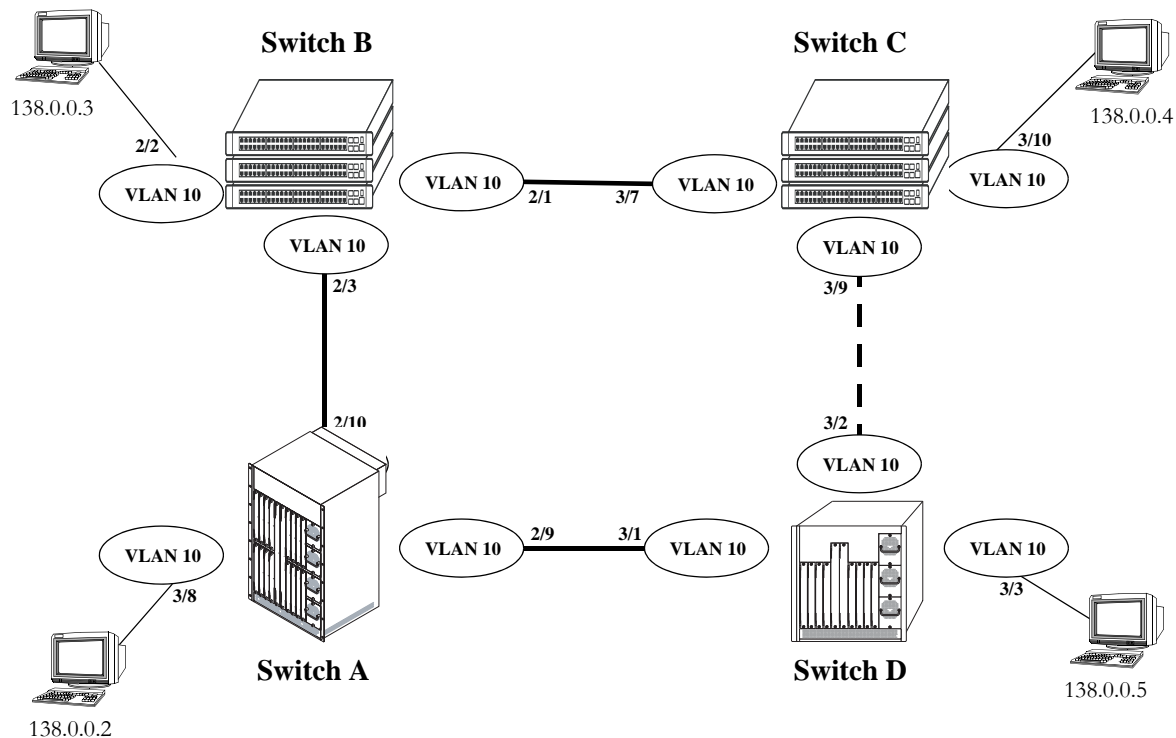
Bridging VLANs Across Multiple Switches

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

- 1 Create a VLAN on each switch with the same VLAN ID number (for example, VLAN 10).
- 2 If using mobile ports for end user device connections, define VLAN rules that classifies mobile port traffic into the VLAN created in Step 1.

- 3 On each switch, assign the ports that provides connections to other switches to the VLAN created in Step 1.
- 4 On each switch, assign the ports that provides connections to end user devices (for example, workstations) to the VLAN created in Step 1. (If using mobile ports, this step occurs 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:



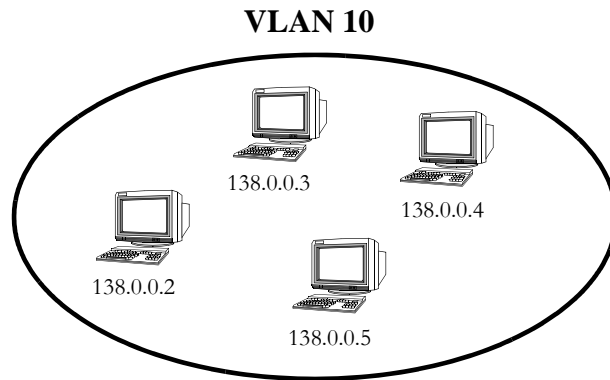
VLAN Bridging Domain: Physical Configuration

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 24, “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 12, “Configuring Spanning Tree,”](#) 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 (for example, 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 AOS Release 6 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. From the GVRP information, a device can continuously update its knowledge of 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 AOS Release 6 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 6350, 6450 |
| Maximum GVRP VLANs | 256 |

GVRP Defaults

The following table lists the defaults for GVRP configuration:

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

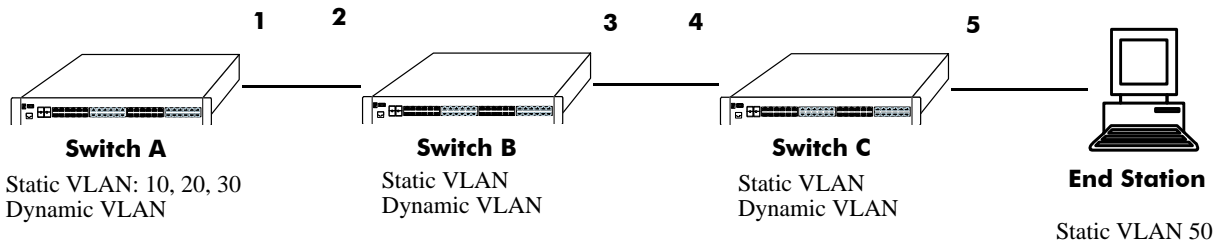
GVRP Overview

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:



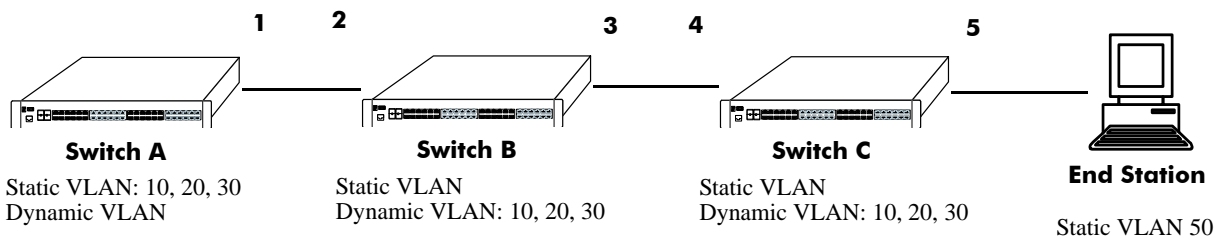
Initial Configuration of GVRP

Switch A has 3 VLANs configured as static VLANs (10, 20, and 30). Other switches on the same network 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:



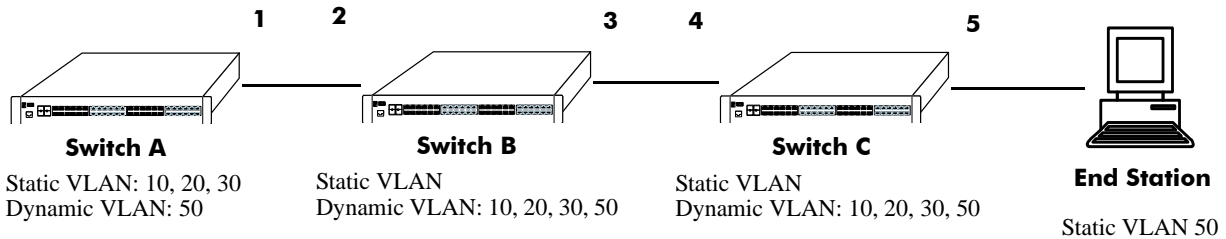
Dynamic Learning of VLANs 10, 20, and 30

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:



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 is 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 is 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 Registration | Restricted Registration | Restricted 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'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 leads 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 have to 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 has 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 discards 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 takes effect only after the GVRP is disabled and enabled again on the switch. If this operation is not done, the VLANs learned earlier is 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 is 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 has to be greater than or equal to thrice the Join timer value (**Leave** \geq **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 is 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 to be dynamically learned on the device, configure the dynamic VLAN registrations by using the **gvrp restrict-vlan-registration** command as follows:

```
-> 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 AOS Release 6 CLI Reference Guide*.

6 Configuring MVRP

Multiple VLAN Registration Protocol (MVRP) is standards-based Layer 2 network protocol for automatic configuration of VLAN information on switches. It was defined in the 802.1ak amendment to 802.1Q-2005.

MVRP provides a method to share VLAN information dynamically and configure the needed VLANs within a layer 2 network. For example, in order to add a switch port to a VLAN, only the end port, or the VLAN-supporting network device connected to the switchport, has to be reconfigured, and all necessary VLAN trunks are dynamically created on the other MVRP-enabled switches. MVRP helps to maintain VLAN configuration dynamically based on current network configurations.

In This Chapter

This chapter describes the MVRP 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 AOS Release 6 CLI Reference Guide*. This chapter provides an overview of MVRP and includes the following information:

- [“Enabling MVRP” on page 6-10](#)
- [“Enabling Transparent Switching” on page 6-11](#)
- [“Configuring the Maximum Number of VLANs” on page 6-11](#)
- [“Configuring MVRP Registration” on page 6-12](#)
- [“Configuring the MVRP Applicant Mode” on page 6-14](#)
- [“Modifying MVRP Timers” on page 6-15](#)
- [“Restricting VLAN Registration” on page 6-16](#)
- [“Restricting Static VLAN Registration” on page 6-16](#)
- [“Restricting VLAN Advertisement” on page 6-17](#)

MVRP Specifications

| | |
|--------------------------|--|
| IEEE Standards Supported | IEEE 802.1ak-2007 Amendment 7: Multiple Registration Protocol IEEEStd802.1Q-2005 Corrigendum 2008 |
| Platforms Supported | OmniSwitch 6350, 6450 |
| Maximum MVRP VLANs | 256 |

MVRP Defaults

The following table lists the defaults for MVRP configuration.

| Parameter Description | Command | Default Value/Comments |
|--|---|---|
| VLAN dynamic registration mode | vlan registration-mode | MVRP |
| Enables or disables MVRP globally on a switch. | mvrp | disabled |
| Enables or disables MVRP on specific ports | mvrp port | disabled |
| Transparent switching | mvrp port | disabled |
| Maximum number of VLANs | mvrp maximum vlan | 256 |
| Registration mode of the port | mvrp registration | normal |
| Applicant mode of the port | mvrp applicant | active |
| Timer value for join timer. | mvrp timer join | <i>600 milliseconds</i> |
| Timer value for leave timer. | mvrp timer leave | <i>1800 milliseconds</i> |
| Timer value for leaveall timer. | mvrp timer leaveall | <i>30000 milliseconds</i> |
| Timer value for periodic timer. | mvrp timer periodic-timer | <i>1 second</i> |
| Restrict dynamic VLAN registration | mvrp restrict-vlan-registration | not restricted |
| Restrict VLAN advertisement | mvrp restrict-vlan-advertisement | not restricted |
| Restrict static VLAN registration | mvrp static-vlan-restrict | By default, ports are assigned to the static VLAN based on MVRP PDU processing. |

Quick Steps for Configuring MVRP

The following steps provide a quick tutorial on how to configure MVRP. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

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

```
-> vlan 5 name "vlan-5"
```

- 2 Assign a port to the VLAN using the **vlan port default** command. For example:

```
-> vlan 5 port default 1/2
```

- 3 Tag the port with one or more VLANs using the **vlan 802.1q** command. For example:

```
-> vlan 7 802.1q 1/2
```

- 4 Enable MVRP globally on the switch by using the **mvrp** command.

```
-> mvrp enable
```

Note. If MVRP is configured, GVRP cannot be configured on that switch and GVRP frames are ignored by the switch.

- 5 Enable MVRP on the port by using the **mvrp port** command. For example, the following command enables MVRP on port 1/2 of the switch:

```
-> mvrp port 1/2 enable
```

- 6 (Optional) Restrict a port from becoming a member of the statically created VLAN by using the **mvrp static-vlan-restrict** command. For example, the following command restricts port 1/5 from becoming a member of static VLAN 10:

```
-> mvrp port 1/5 static-vlan-restrict vlan 10
```

Note. To view the global configuration details of the router, enter the **show mvrp configuration** command. The globally configured details are displayed as shown:

```
-> show mvrp configuration
MVRP Enabled : yes,
Transparent Switching Enabled: no,
Maximum VLAN Limit : 256
```

To view the MVRP configuration for a specific port, enter the **show mvrp port** command. The configuration data of the particular port is displayed as shown:

```
-> show mvrp port 1/2
Port 1/2:
MVRP Enabled : no,
Registrar Mode : normal,
Applicant Mode : participant,
Join Timer (msec) : 600,
Leave Timer (msec) : 1800,
LeaveAll Timer (msec) : 30000,
```

```
Periodic Timer (sec) : 1,  
Periodic Tx Status  : disabled
```

See the *OmniSwitch AOS Release 6 CLI Reference Guide* for information about the fields in this display.

MRP Overview

Multiple Registration Protocol (MRP) was introduced as a replacement for GARP with the IEEE 802.1ak-2007 amendment. The Multiple VLAN Registration Protocol (MVRP) defines a MRP Application that provides the VLAN registration service.

MVRP provides a mechanism for dynamic maintenance of the contents of dynamic VLAN registration Entries for each VLAN, and for propagating the information they contain to other bridges. This information allows MVRP-aware devices to dynamically establish and update their knowledge of the set of VLANs that currently have active members, and through which ports those members can be reached. The main purpose of MVRP is to allow switches to automatically discover some of the VLAN information that would otherwise need to be manually configured.

MVRP Overview

MVRP acts as an MRP application, sending and receiving MVRP information encapsulated in an ethernet frame on a specific MAC address. MVRP allows both end stations and bridges in a bridged local area network to issue and revoke declarations relating to membership of VLANs. Each MVRP device that receives the declaration in the network creates or updates a dynamic VLAN registration entry in the filtering database to indicate that the VLAN is registered on the reception port.

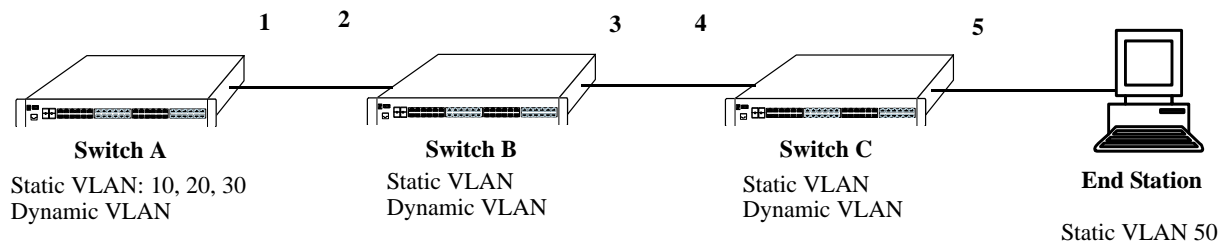
In this way, MVRP provides a method to share VLAN information within a layer 2 network dynamically, and configure the needed VLANs. For example, in order to add a switch port to a VLAN, only the end port, or the VLAN-supporting network device connected to the switchport, need be reconfigured, and all necessary VLAN trunks are dynamically created on the other MVRP-enabled switches. Without using MVRP, either a manual configuration of VLAN trunks or use of a manufacturer-specific proprietary method is necessary. In short, MVRP helps to maintain VLAN configuration dynamically based on current network configurations.

How MVRP Works

An MVRP enabled port sends MRPDUs advertising the VLAN enabling another MVRP aware port receiving advertisements over a link to join the advertised VLAN dynamically. All ports of a dynamic VLAN operate as tagged ports for that VLAN.

An MVRP enabled port can forward an advertisement for a VLAN it learned about from other ports on the same switch. However, the forwarding port does not join that VLAN on its own until an advertisement for that VLAN is received on that same port.

The following example illustrates the VLAN advertisements and Dynamic Joining.



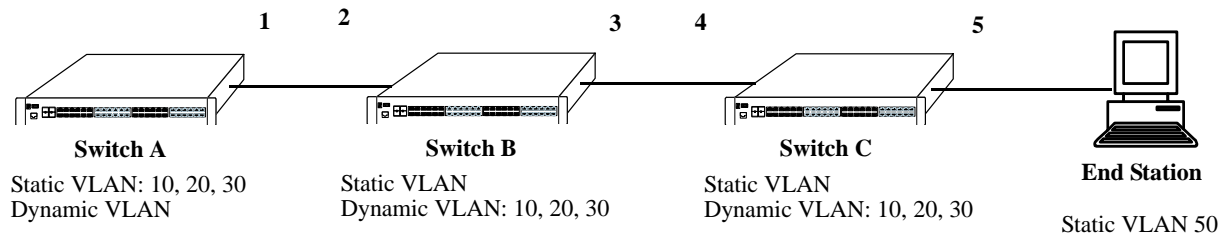
Initial Configuration of MVRP

Switch A has 3 VLANs configured as static VLANs (10, 20, and 30). Other switches on the same network 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. All the ports are in the same Spanning tree instance and are in forwarding state. Hence, as the [Initial Configuration of MVRP](#) diagram 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 B 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 Switch C 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 configuration sequence of advertisements and registration of VLANs results in the following configuration.

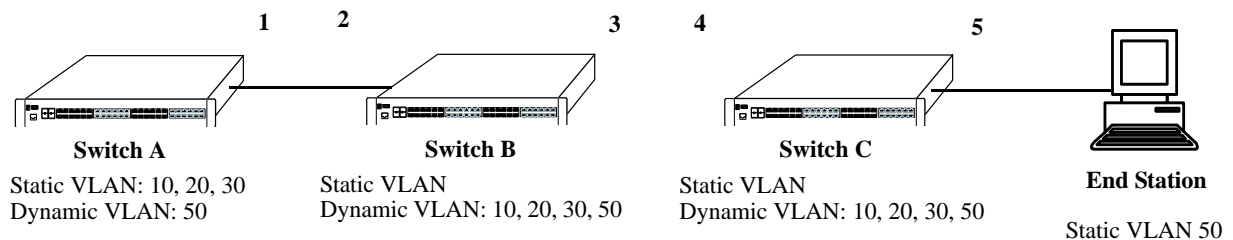


Dynamic Learning of VLANs 10, 20, and 30

Here, the end station advertises itself as a member of VLAN 50. As the [Dynamic Learning of VLANs 10, 20, and 30](#) 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 as follows:



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.

Interaction With Other Features

This section contains important information about how other OmniSwitch features interact with MVRP. Refer to the specific chapter for each feature to get more detailed information about how to configure and use the feature.

GVRP

If MVRP is configured, GVRP cannot be configured and the GVRP frames are ignored on that switch. MVRP is functionally independent of the GVRP.

When the device has legacy GVRP commands in the boot.cfg (for example, during image upgrade from a previous release which does not support MVRP) and the default mode is configured for MVRP, the GVRP commands are still accepted and the VLAN registration mode is internally changed to GVRP.

There is an option to change the operational mode between MVRP and GVRP. But, when you change the mode, it results in the complete deletion of static as well as dynamic configurations of the existing operational mode.

STP

MVRP feature is supported only in STP flat mode. If MVRP is configured in the system with STP flat mode, then STP mode cannot be changed to 1x1 mode. When a topology change is detected by STP, MAC addresses for the dynamic VPAs learned by MVRP is also deleted.

IPM VLAN

MVRP is not supported on IP Multicast VLANs (IPMVLANs). If MVRP PDU for IPMVLAN registration is received on standard/network port, the PDUs are discarded. IPMVLAN is not advertised by MVRP.

Configuring MVRP

This section describes how to configure MVRP using the Command Line Interface (CLI) commands.

Enabling MVRP

MVRP is used primarily to prune unnecessary broadcast and unknown unicast traffic, and dynamically create and manage VLANs. MVRP has to be globally enabled on a switch before it can start forwarding MVRP frames. When MVRP is configured on a switch, GVRP cannot be configured on that switch and when a port is enabled for MVRP, it cannot be converted as a mobile, mirroring, aggregate, VPLS Access, or a VLAN stacking User port.

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

```
-> mvrp enable
```

To disable MVRP globally on the switch, use disable option of the **mvrp** command as shown:

```
-> mvrp disable
```

Note. Disabling MVRP globally leads to the deletion of all learned VLANs.

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

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

```
-> mvrp port 1/2 enable
```

Similarly, to enable MVRP on aggregate group 10, enter:

```
-> mvrp linkagg 10 enable
```

To disable MVRP on a specific port, use disable option of the **mvrp port** command as shown:

```
-> mvrp port 1/2 enable
```

Note. MVRP can be configured only on fixed, 802.1 Q and aggregate ports. It cannot be configured on mirror, aggregate, mobile, VPLS Access, and VLAN Stacking User ports.

Enabling Transparent Switching

A switch in the MVRP transparent mode floods MVRP frames to other switches transparently when MVRP 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 MVRP is globally enabled on a switch, transparent switching does not have any effect on the switch.

You can configure the switch to propagate MVRP frames transparently using the **mvrp port** command, as shown:

```
-> mvrp transparent-switching enable
```

Use the disable option of this command to disable the transparent switching capability of the switch. For example:

```
-> mvrp transparent-switching disable
```

Note. When both MVRP and MVRP transparent switching are globally disabled, the switch discards the MVRP frames.

Configuring the Maximum Number of VLANs

A switch can create dynamic VLANs using MVRP. By default, the maximum number of dynamic VLANs that can be created using MVRP is 256. 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 MVRP is disabled and enabled again on the switch. If this operation is not done, the VLANs learned earlier are maintained.

To modify the maximum number of dynamic VLANs the switch is allowed to create, use the **mvrp maximum vlan** command as shown:

```
-> mvrp maximum vlan 150
```

Configuring MVRP Registration

MVRP 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, thereby preventing attempts to send data to devices that are not reachable.

The following sections describe MVRP registration on switches:

Setting MVRP 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 **mvrp registration** command. For example, to configure port 2 of slot 1 in normal mode, enter the following:

```
-> mvrp port 1/2 registration normal
```

To view the registration mode of the port, use the **show mvrp port** command. For example:

```
-> show mvrp port 1/2

MVRP Enabled           : no,
Registrar Mode         : normal,
Applicant Mode         : participant,
Join Timer (msec)      : 600,
Leave Timer (msec)      : 1800,

LeaveAll Timer (msec)   : 30000,
Periodic Timer (sec)   : 1,
Periodic Tx status     : disabled
```

Setting MVRP 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 **mvrp registration** command. For example, to configure port 2 of slot 1 to fixed mode, enter the following:

```
-> mvrp port 1/2 registration fixed
```

To view the registration mode of the port, use the **show mvrp port** command. For example,

```
-> show mvrp port 1/2

MVRP Enabled           : no,
Registrar Mode         : fixed,
Applicant Mode         : participant,
Join Timer (msec)      : 600,
Leave Timer (msec)      : 1800,
LeaveAll Timer (msec)   : 30000,
Periodic Timer (sec)   : 1,
Periodic Tx status     : disabled
```

Note. The registration mode for the default VLANs of all the ports in the switch is set to normal.

Setting MVRP Forbidden Registration

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

To configure a port to forbidden mode, use the **mvrp registration** command. For example, to configure port 2 of slot 1 to forbidden mode, enter the following:

```
-> mvrp port 1/2 registration forbidden
```

To view the registration mode of the port, use the **show mvrp port** command. For example,

```
-> show mvrp port 1/2

MVRP Enabled           : no,
Registrar Mode         : forbidden,
Applicant Mode         : participant,
Join Timer (msec)      : 600,
Leave Timer (msec)      : 1800,
LeaveAll Timer (msec)   : 30000,
Periodic Timer (sec)   : 1,
Periodic Tx status     : disabled
```

To view the MVRP configurations for all the ports, including timer values, registration and applicant modes, enter the following:

```
-> show mvrp port enabled
```

| Port | Join Timer (msec) | Leave Timer (msec) | LeaveAll Timer (msec) | Periodic Timer (sec) | Registration Mode | Applicant Mode | Periodic Tx Status |
|------|-------------------------|--------------------------|-----------------------------|----------------------------|----------------------|-------------------|--------------------|
| 1/1 | 600 | 1800 | 30000 | 2 | fixed | active | enabled |

| | | | | | | | |
|------|-----|------|-------|---|-------|--------|---------|
| 1/2 | 600 | 1800 | 30000 | 2 | fixed | active | enabled |
| 1/7 | 600 | 1800 | 30000 | 2 | fixed | active | enabled |
| 1/8 | 600 | 1800 | 30000 | 2 | fixed | active | enabled |
| 2/24 | 600 | 1800 | 30000 | 2 | fixed | active | enabled |

Configuring the MVRP Applicant Mode

The MVRP applicant mode determines whether MVRP PDU exchanges are allowed on a port, depending on the Spanning Tree state of the port. This mode can be configured to be **participant**, **nonparticipant**, or **active**. By default, the port is in the participant mode.

To prevent undesirable Spanning Tree Protocol topology reconfiguration on a port, configure the MVRP applicant mode as active. Ports in the MVRP active applicant state send MVRP VLAN declarations even when they are in the STP blocking state, thereby preventing 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 **mvrp applicant** command. For example, to set the applicant mode of port 1/2 to active, enter the following:

```
-> mvrp port 1/2 applicant active
```

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

To set the applicant mode of port 1/2 to participant mode, enter the following:

```
-> mvrp port 1/2 applicant participant
```

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

To set the applicant mode of port 1/2 to non-participant mode, enter the following:

```
-> mvrp port 1/2 non-participant
```

The applicant mode of the port can be set to the default value by using the **mvrp applicant** command. To set the MVRP applicant mode of port 1/2 to the default mode (active mode), enter the following command:

```
-> mvrp port 1/2 active
```


Modifying MVRP Timers

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

- **Join** timer—The maximum time an MVRP 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 an MVRP instance takes to generate LeaveAll messages. The LeaveAll message instructs the port to modify the MVRP state of all its VLANs to **Leave**.
- **Periodic** timer—The time frequency with which the messages are transmitted again and again.

The default values of the Join, Leave, and LeaveAll timers are 600 ms, 1800 ms, and 30000 ms, respectively.

When you set the timer values, the value for the Leave timer must be greater than or equal to twice the Join timer value plus 100 milliseconds. ($\text{Leave} \geq \text{Join} * 2 + 100$). The LeaveAll timer value must be greater than or equal to the Leave timer value ($\text{LeaveAll} \geq \text{Leave}$). If you attempt to set a timer value that does not adhere to these rules, an error message is displayed.

For example, if you set the Leave timer to 1700 ms and attempt to configure the Join timer to 400 ms, an error is returned. Set the Leave timer to at least 1800 ms and then set the Join timer to 600 ms.

To modify the Join timer value, use the **mvrp timer join** command. For example, to modify the Join timer value of port 1/2, enter the following:

```
-> mvrp port 1/2 timer join 600
```

The Join timer value of port 1/2 is now set to 600 ms.

To set the Leave timer value of port 1/2 to 1800 ms, enter the command as shown:

```
-> mvrp port 1/2 timer leave 1800
```

To set the LeaveAll timer of port 1/2 to 30000 ms, enter the command as shown:

```
-> mvrp port 1/2 timer leaveall 30000
```

To set the Periodic timer of port 1/2 to 1 second, enter the command as shown:

```
-> mvrp port 1/2 timer periodic-timer 1
```

To view the timer value assigned to a particular port, use the **show mvrp timer** command.

```
-> show mvrp port 1/2 timer

Join Timer (msec)       : 600,
Leave Timer (msec)       : 1800,
LeaveAll Timer (msec)    : 30000,
Periodic-Timer (sec)    : 1
```

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

Restricting VLAN Registration

Restricted VLAN registration restricts MVRP from dynamically registering specific VLAN or VLANs 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 **mvrp restrict-vlan-registration** command as shown:

```
-> mvrp port 1/1 restrict-vlan-registration vlan 4
```

Here, VLAN 4 cannot be learned by the device dynamically. However, if the VLAN 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 **mvrp restrict-vlan-registration** command as shown:

```
-> no mvrp port 1/1 restrict-vlan-registration vlan 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 **mvrp static-vlan-restrict** command as shown:

```
-> mvrp port 1/9 static-vlan-restrict vlan 5
```

Note. This command does not apply to dynamic VLANs.

Here, the port 1/9 is restricted from becoming a MVRP member of VLAN 5.

To restrict a port from becoming a member of a range of statically created VLANs, enter the **mvrp static-vlan-restrict** command as shown:

```
-> mvrp port 1/9 static-vlan-restrict vlan 5-9
```

Here, port 1/9 is restricted from becoming a MVRP 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 **mvrp static-vlan-restrict** command. To allow port 1/2 to become a member of a statically created VLAN, enter the command as shown:

```
-> no mvrp port 1/2 static-vlan-restrict vlan 5
```

Restricting VLAN Advertisement

VLANs learned by a switch through MVRP 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 `mvrp restrict-vlan-advertisement` command to restrict the propagation of VLAN information on a specified port as shown:

```
-> mvrp port 1/1 restrict-vlan-advertisement vlan 5
```

Here, VLAN 5 is not allowed to propagate on port 1 of slot 1.

To enable the propagation of dynamic VLANs on the specified port, use the no form of the command. To restrict VLAN 5 from being propagated to port 1/1, enter the command as shown:

```
-> no mvrp port 1/1 restrict-vlan-advertisement vlan 5
```

Verifying the MVRP Configuration

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

| | |
|------------------------------------|---|
| show mvrp last-pdu-origin | Displays the source MAC address of the last MVRP message received on specific ports or aggregates. |
| show mvrp configuration | Displays the global configuration for MVRP. |
| show mvrp linkagg | Displays the MVRP configuration for a specific port or an aggregate of ports. |
| show mvrp port | Displays the MVRP configurations for all the ports, including timer values, registration and applicant modes. |
| show mvrp vlan-restrictions | Displays the list of VLANS learned through MVRP and their details. |
| show mvrp timer | Displays the timer values configured for all the ports or a specific port. |
| show mvrp statistics | Displays the MVRP statistics for all the ports, aggregates, or specific ports. |
| mvrp clear-statistics | Clears MVRP statistics for all the ports, an aggregate of ports, or a specific port. |

For more information about the output details that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

7 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 7-4.](#))
- 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 24, “Configuring 802.1Q.”](#))
- Configuring ports as members of a link aggregate that is assigned to a configured default VLAN. (See [Chapter 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 7-5](#) 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Statically assigning ports to VLANs on [page 7-4.](#)
- Dynamically assigning ports to VLANs (port mobility) [page 7-10.](#)
- Configuring mobile port properties (including authentication) on [page 7-16.](#)

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 6350, 6450 |
| 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 | <code>vlan port default</code> | All ports initially associated with default VLAN 1. |
| Port mobility | <code>vlan port mobile</code> | Disabled |
| Bridge mobile port traffic that doesn't match any VLAN rules on the configured default VLAN | <code>vlan port default vlan</code> | Disabled |
| Drop mobile port dynamic VLAN assignments when learned mobile port traffic that triggered the assignment ages out | <code>vlan port default vlan restore</code> | Enabled |
| Enable Layer 2 authentication on the mobile port | <code>vlan port authenticate</code> | Disabled |
| Enable 802.1x port-based access control on a mobile port | <code>vlan port 802.1x</code> | Disabled |

Sample VLAN Port Assignment

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

- 1 Create VLAN 255 with a description (for example, 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 does 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 7-5](#) 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 24, “Configuring 802.1Q,”](#) for more information.)
- Configuring ports as members of a link aggregate that is assigned to a configured default VLAN. (See [Chapter 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 7-18.](#)

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 7-18](#).

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 7-5](#) for more information.)
- Packet contents matches criteria defined in a VLAN rule. (See [“VLAN Rule Classification” on page 7-8](#) 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 24, “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

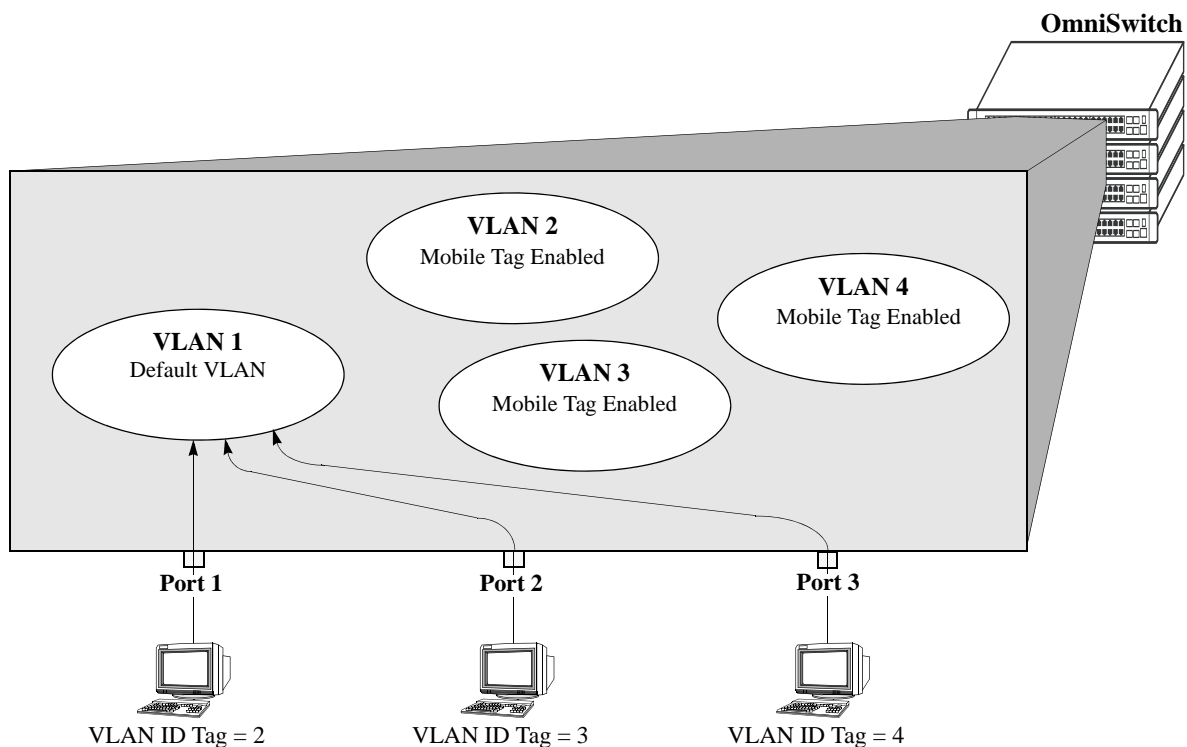
mobile tagging enabled, the port is dynamically assigned to the mobile tag VLAN and not the matching rule VLAN.

- 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,

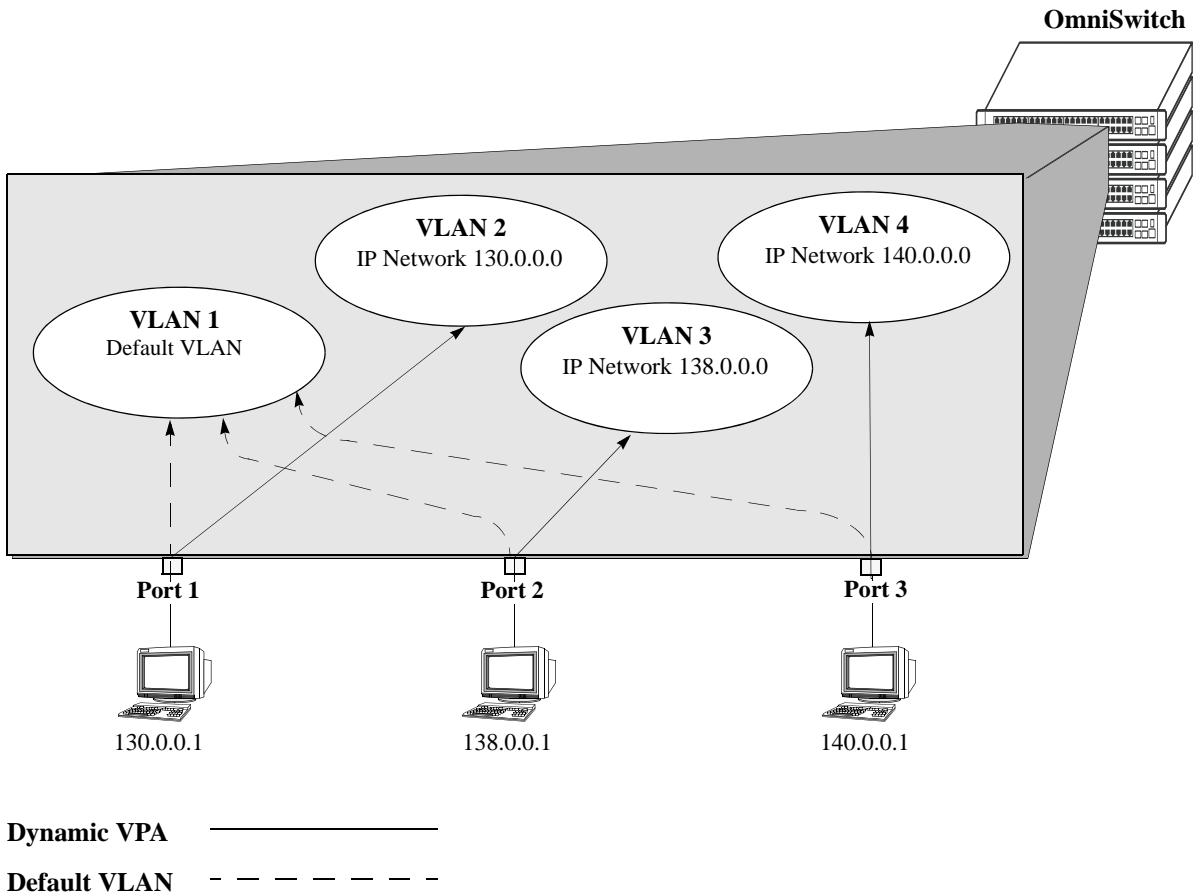
- All three ports have workstations that are configured to send packets with an 802.1Q VLAN ID tag for three different VLANs (VLAN 2, 3, and 4).
- Mobility is enabled on each of the workstation ports.
- VLAN 1 is the configured default VLAN for each port.
- VLANs 2, 3, and 4 are configured on the switch, 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 7-7](#) 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 9, “Defining VLAN Rules,”](#) for more information).

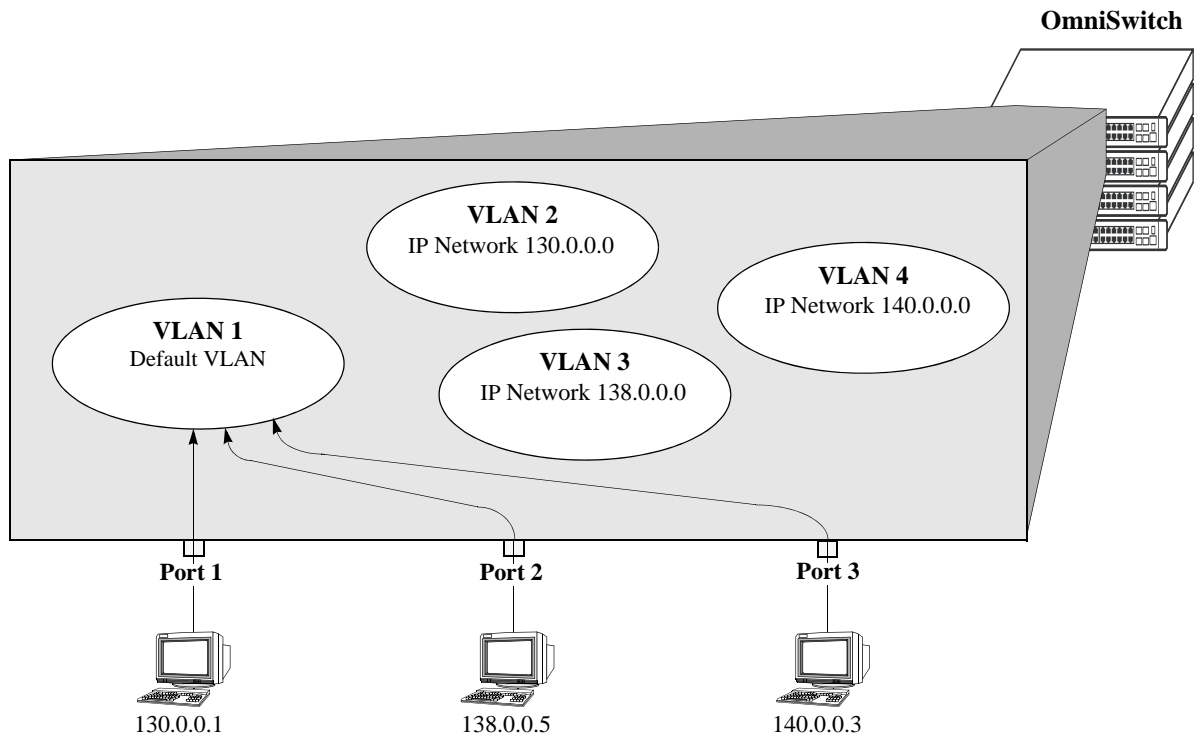
Note the following items when using VLAN rule classification:

- 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 9, “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 7-9](#),

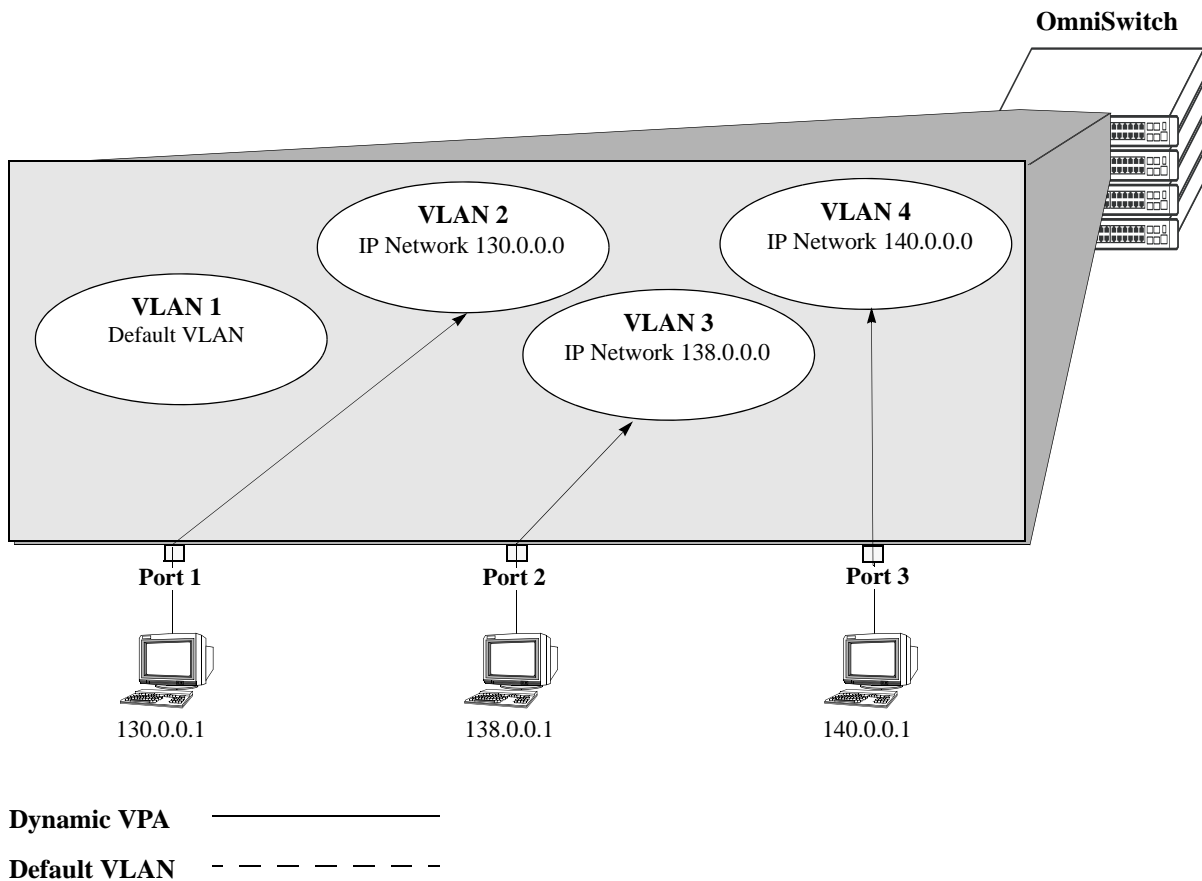
- All three ports have workstations that belong to three different IP subnets (130.0.0.0, 138.0.0.0, and 140.0.0.0).
- Mobility is enabled on each of the workstation ports.
- VLAN 1 is the configured default VLAN for each port.
- Three additional VLANs are configured on the switch, each one has an IP 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 7-10](#) 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.



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 participates in dynamic VLAN assignment. See [“Enabling/Disabling Port Mobility” on page 7-11](#) for detailed procedures.
- 2** Enable/disable mobile port properties that determine mobile port behavior. See [“Configuring Mobile Port Properties” on page 7-16](#) for detailed procedures.
- 3** Create VLANs that receives 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 triggers dynamic assignment of a mobile port to the VLANs created in Step 3. See [“VLAN Rule Classification” on page 7-8](#) and [“VLAN Mobile Tag Classification” on page 7-5](#) 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 7-5](#) 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 7-11](#) 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 39, “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 AOS Release 6 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 44, “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, restores 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 does 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 (for example, 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 7-14](#).
- How Mobile Port VLAN Assignments Age on [page 7-15](#).

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 7-4](#).

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 7-18](#).

What is a Secondary VLAN?

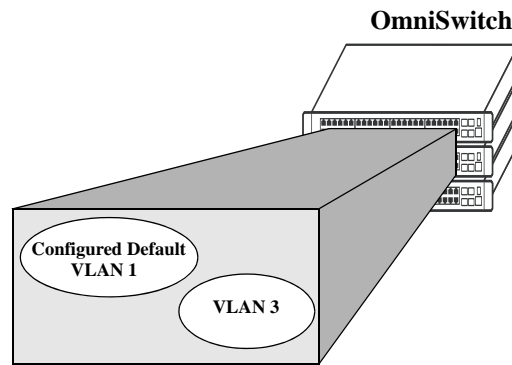
All mobile ports start out with a configured default VLAN assignment. When mobile port traffic matches VLAN criteria, the port is assigned to that VLAN. Secondary VLANs are any VLAN a port is subsequently assigned to that is not the configured default VLAN for that port.

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 7-18.

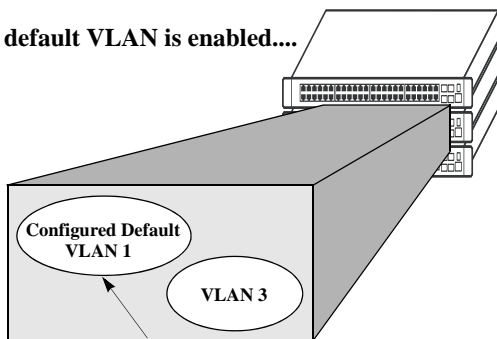


Device connected to a mobile port sends traffic. If the traffic matches existing VLAN criteria, then the mobile port and its traffic are dynamically assigned to that VLAN.



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....



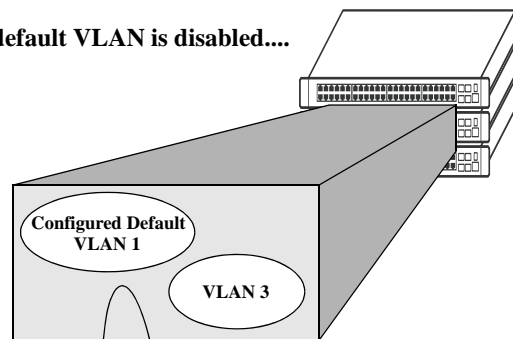
Device traffic that does not match any VLAN rules is forwarded on the mobile port's configured default VLAN.



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.

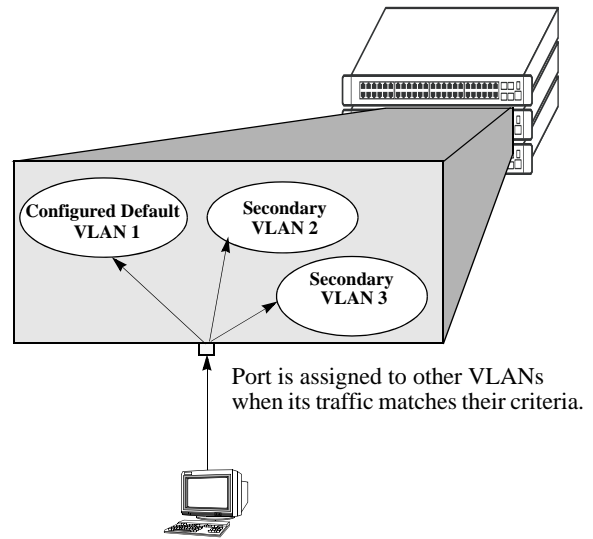
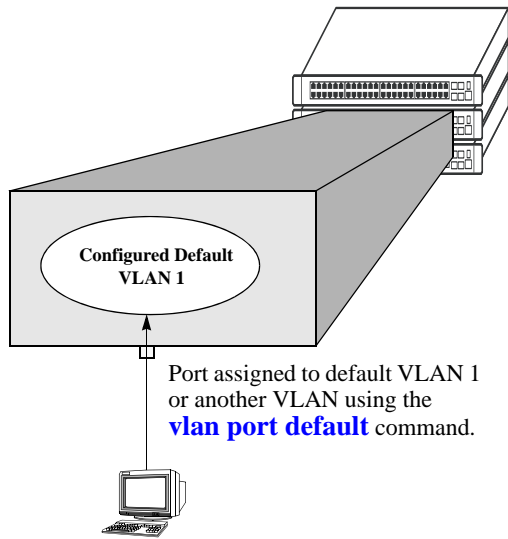


Why disable default VLAN?

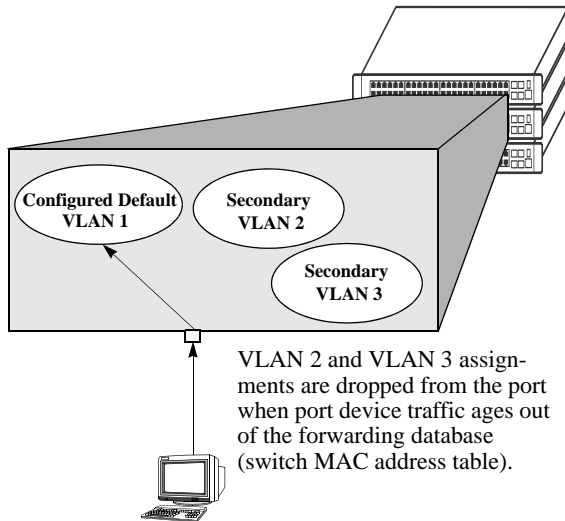
Reduces unnecessary traffic flow on a port's configured default VLAN.

Restricts dynamic assignment to mobile port traffic that matches one or more VLAN rules.

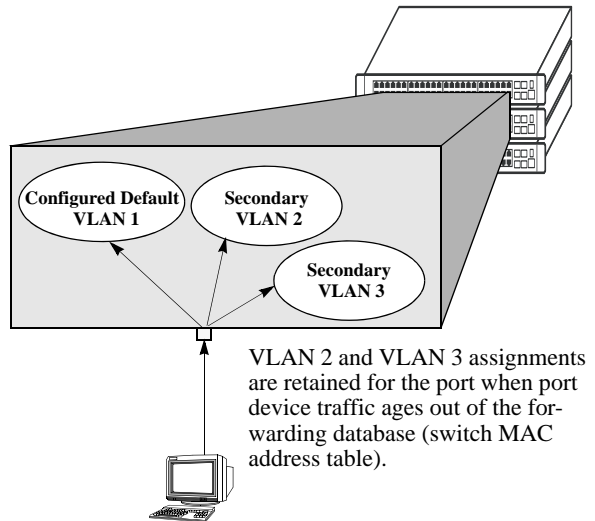
How Mobile Port Traffic that Does Not Match any VLAN Rules is Classified



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 (for example, 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 AOS Release 6 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 7-18](#) 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 (for example, mobile ports with default VLAN enabled or non-mobile, fixed ports).

See [“Understanding Mobile Port Properties” on page 7-12](#) 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 9, “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 7-12 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 switches. For more information, see [Chapter 37, “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 43, "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 7-3](#). For more information about the resulting display from this command, see the *OmniSwitch AOS Release 6 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

      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 7-3](#). For more information about the resulting display from this command, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

8 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 8-3 or [“Deleting a Port Mapping Session”](#) on page 8-3.
- [Enabling/Disabling a Port Mapping Session](#)—see [“Enabling a Port Mapping Session”](#) on page 8-4 or [“Disabling a Port Mapping Session”](#) on page 8-4.
- [Configuring a Port Mapping Direction](#)—see [“Configuring Unidirectional Port Mapping”](#) on page 8-4 and [“Restoring Bidirectional Port Mapping”](#) on page 8-4.
- [Configuring an example Port Mapping Session](#)—see [“Sample Port Mapping Configuration”](#) on page 8-5.
- [Verifying a Port Mapping Session](#)—see [“Verifying the Port Mapping Configuration”](#) on page 8-6.

Port Mapping Specifications

| | |
|---------------------|---|
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 `port mapping dynamic-proxy-arp` 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 is 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 (that is, bidirectional), enter **port mapping** followed by the session ID and **bidirectional**. For example, to restore the direction (that is, 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

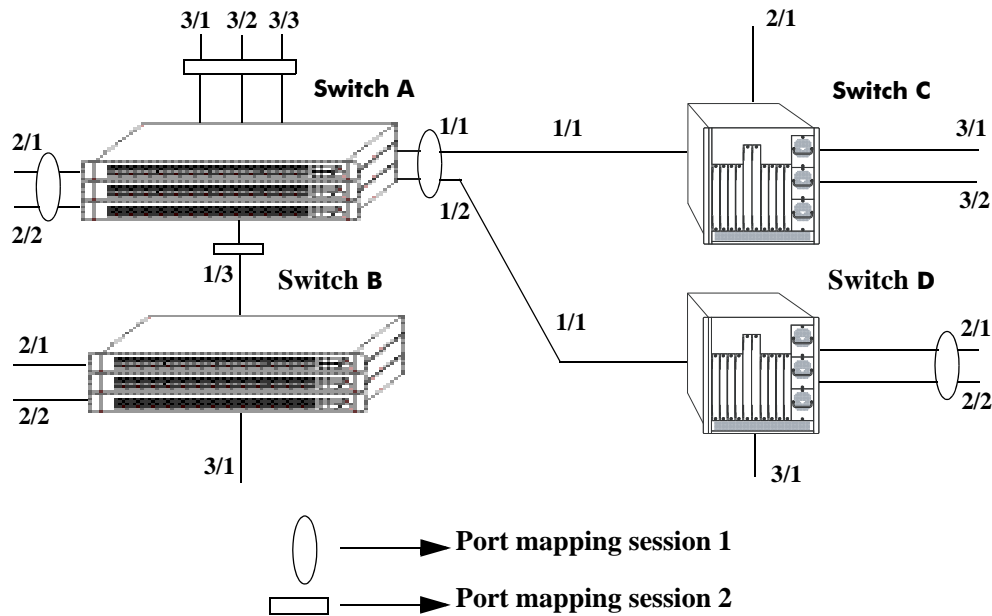
This section provides an example port mapping network configuration. In addition, a tutorial is also included that provides steps on how to configure the example port mapping session using the Command Line Interface (CLI).

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 8-5](#).

- 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:

port mapping dynamic-proxy-arp 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 AOS Release 6 CLI Reference Guide*.

9 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 AOS Release 6 CLI Reference Guide*. Refer to [Chapter 4, “Configuring VLANs,”](#) and [Chapter 7, “Assigning Ports to VLANs,”](#) for information about the VLAN mobile tagging feature.

Configuration procedures described in this chapter include:

- Defining DHCP rules on [page 9-9](#).
- Defining MAC address rules on [page 9-10](#).
- Defining IP network address rules on [page 9-11](#).
- Defining protocol rules on [page 9-12](#).
- Defining forwarding-only port rules on [page 9-13](#).
- Verifying the VLAN rule configuration on [page 9-17](#).

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 7, “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 6350, 6450 |
| 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 • 96 MAC and 16 IP rules (OS6350) • 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 AOS Release 6 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 creates 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 (for example, Finance IP Network) using the following command:

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

2 Define an IP network address rule for VLAN 255 that captures 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 captures 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 7, “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 9-8](#) for instructions on how to create a VLAN rule.

The type of rule defined determines the type of traffic that triggers a dynamic port assignment to the VLAN and the type of traffic the VLAN forwards 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 9-5 |
| MAC Address MAC Address Range | “MAC Address Rules” on page 9-5 |
| Network Address | “Network Address Rules” on page 9-5 |
| Protocol | “Protocol Rules” on page 9-5 |
| Port | “Port Rules” on page 9-6 |

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 AOS Release 6 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 does not contain an entry for the DHCP source MAC address. The [show vlan port](#) command output, however, contains 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 consists 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 7, "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 7, "Assigning Ports to VLANs,"](#) for more information about VLAN mobile tag classification.

The VLAN rule precedence table on [page 9-7](#) 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 subnet 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 9-3](#) 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 9-6](#) 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 9-9 |
| DHCP MAC Range | “Defining DHCP MAC Range Rules” on page 9-9 |
| DHCP Port | “Defining DHCP Port Rules” on page 9-10 |
| DHCP Generic | “Defining DHCP Generic Rules” on page 9-10 |
| MAC Address | “Defining MAC Address Rules” on page 9-10 |
| MAC Address Range | “Defining MAC Range Rules” on page 9-11 |
| Network Address | “Defining IP Network Address Rules” on page 9-11 and “Defining Protocol Rules” on page 9-12 |

| Rule | See |
|----------|--|
| Protocol | “Defining Protocol Rules” on page 9-12 |
| Port | “Defining Port Rules” on page 9-13 |

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 AOS Release 6 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 9-14](#) 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 (for example, 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 9-14](#) 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 (for example, 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 9-14](#) 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 (for example, 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 creates 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

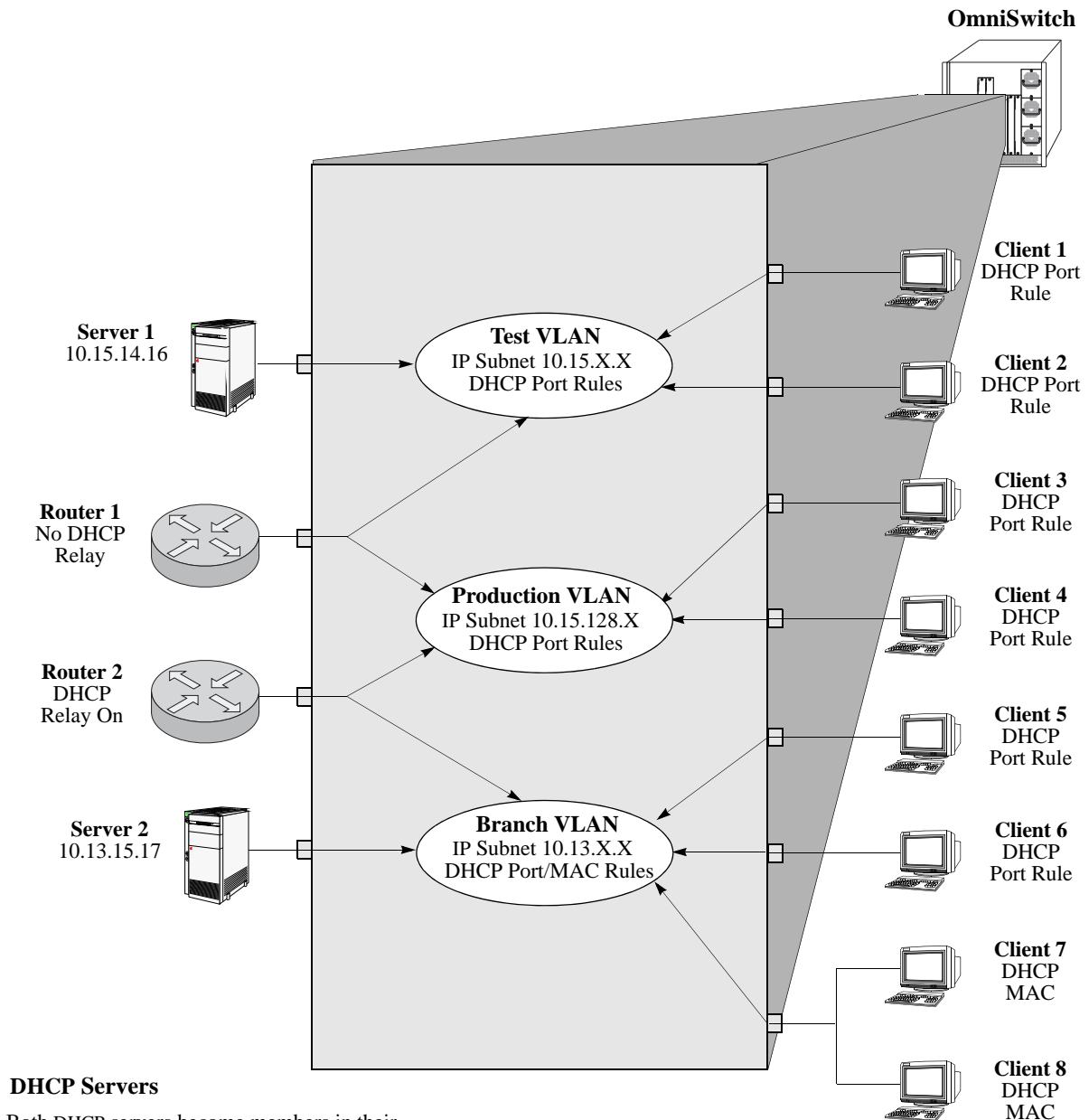
DHCP clients must communicate with a DHCP server at initialization. The most reliable way to ensure this communication is for the server and its associated clients to share the same VLAN. However, if the network configuration does not lend itself to this solution (as the Production VLAN does not in this 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 AOS Release 6 CLI Reference Guide*. An example of the output for the **show vlan rules** command is also given in [“Sample VLAN Rule Configuration” on page 9-3](#).

10 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 to support multiple customer sites or networks distributed over the edges of a service provider network.

Standard VLAN support on NNI ports' allows any standard (non-service) VLAN to be associated to NNI ports of type untagged or 802.1q tagged. However, VLAN 1, cannot be associated as untagged member to a NNI port. 802.1q services, QinQ service and untagged services can be configured using the same uplink NNI port. This allows the customer to use an untagged management VLAN to manage the switch through NNI ports.

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.
- Built in UNI profiles IEEE-FWD-ALL and IEEE-DROP-ALL to tunnel or discard all IEEE multicast MAC addresses traffic associated to UNI port.
- Multiple TPIDs (0x8100, 0x88a8 & 0x9100) supported and interpreted on UNI ports.
- L2 control frames with up to 8 VLAN tag headers are accepted and on UNI ports.
- Custom L2 protocol entry for proprietary protocol with multicast MAC addresses for specific packet control.
- 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 AOS Release 6 CLI Reference Guide*.

This chapter provides an overview of VLAN Stacking and includes the following topics:

- “VLAN Stacking Specifications” on page 10-3.
- “VLAN Stacking Defaults” on page 10-3.
- “VLAN Stacking Overview” on page 10-4.
- “Interaction With Other Features” on page 10-10.
- “Configuring VLAN Stacking Services” on page 10-14
- “Configuring Custom L2 Protocol” on page 10-26
- “Configuring MAC-Tunneling for SVLAN” on page 10-29
- “VLAN Stacking Application Examples” on page 10-30.
- “Wire-Speed Ethernet Loopback Test” on page 10-33.
- “Control Protocol Tunneling Frame Statistics” on page 10-27
- “Control HW Tunneling” on page 10-28

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 6450 |
| Maximum number of SVLANs | 4093 (VLAN 2 through 4094) |
| Maximum number of UNI port associations with CVLANs. | 128 |
| Maximum number of custom L2 protocol entries | 64 |
| Maximum number of custom L2 protocol entries associated per UNI profile | 16 |
| Maximum number of VLAN tags accepted from the incoming dataframe header: | |
| - Preserve Mode | 7 |
| -Translate Mode | 8 |
| Maximum number of NNI TPID values that can be configured | 3 (other than 0x8100, 0x9100 & 0x88a8) |
| 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 |

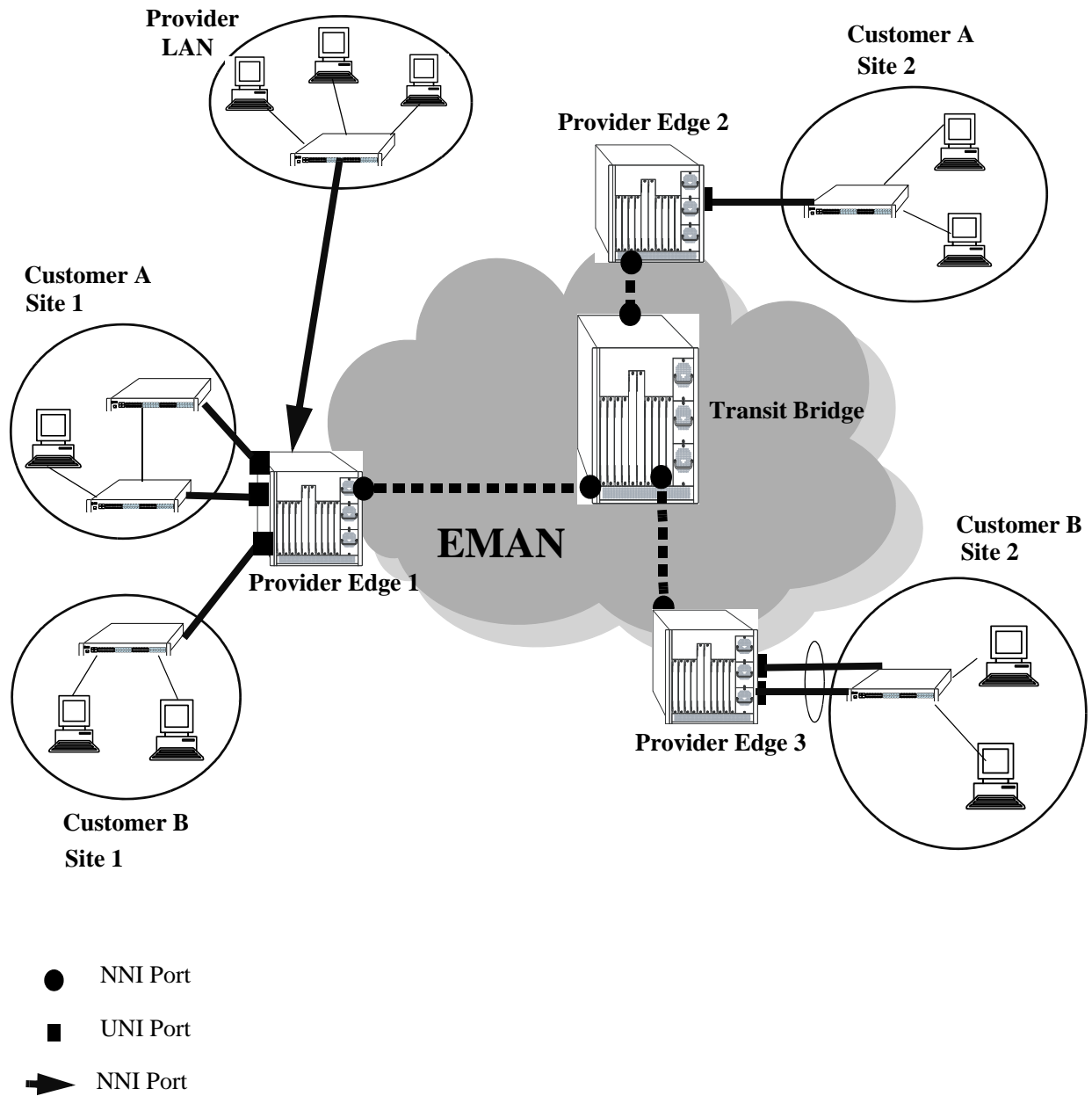
| Parameter Description | Command | Default Value/Comments |
|--|---|--|
| Treatment of customer protocol control frames ingressing on a VLAN Stacking user port. | ethernet-service uni-profile | <p>Processed Frames: 802.3ad, UDLD, OAM, LACP Marker</p> <p>Tunneled Frames: STP, GVRP, MVRP</p> <p>Discarded Frames: 802.1x, 802.1ab, AMAP, VTP VLAN, Uplink Fast, PVST, PAGP, DTP, CDP</p> |
| Treatment of L2 protocol control frames having a destination mac-address of 01-80-C2-00-00-XX after associating a VLAN Stacking UNI profile with a UNI port. | ethernet-service uni uni-profile | <p>ieee-fwd-all: forward all frames as normal data without mac tunneling.</p> <p>ieee-drop-all: discard all such frames</p> |

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 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. If the switch transports packets between a customer-facing port and a network port or between two customer-facing ports, it is considered a PE bridge.
- **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. The same switch can function as 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 can consist of a single port or an aggregate of ports, and can accept tagged or untagged traffic.

The following illustration shows how VLAN Stacking is used to tunnel customer traffic through a service provider network:



VLAN Stacking Elements

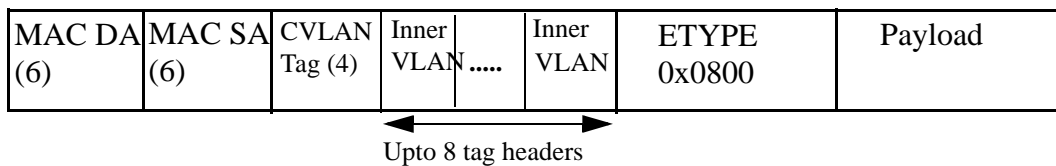
How VLAN Stacking Works

On the Provider Edge bridge (PE), a unique tunnel (SVLAN) ID is assigned to each customer. When the tunnel (SVLAN) is created on the bridge, the VLAN manager and VLAN stacking software store this SVLAN ID. For example, an SVLAN with ID 100 from the provider bridge VLAN tunnels the customer VLAN traffic associated with tunnel 100. 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 bridge port and/or VLANs of the customer provider. 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.

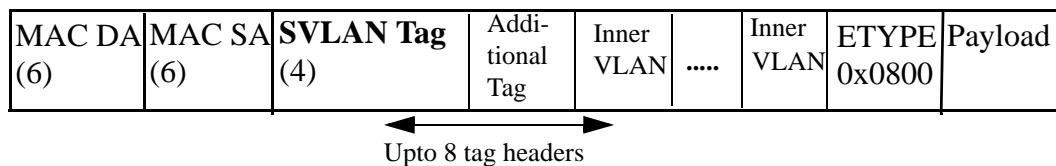
The following provides an example of how a packet ingressing on a VLAN Stacking UNI port that is tagged with the customer VLAN (CVLAN) ID transitions through the VLAN Stacking encapsulation process:

- 1 Packet with CVLAN tag ingressing on a user port.



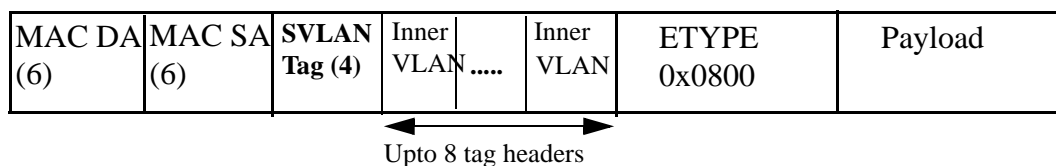
Note. MAC processing and tunneling is supported for up to 8 VLAN tag headers at a UNI port. Similarly, MAC processing and tunneling is supported for up to 8 VLAN tag headers at an NNI port.

- 2 **Double Tagging** inserts the SVLAN tag in the packet. The packet is sent out the network port with double tags (SVLAN+Additional tag for SVLAN).



Note. Double tagging is applied when **preserve** mode is configured using the **ethernet-service sap profile** command.

- 3 **VLAN Translation** replaces the CVLAN Tag with SVLAN Tag. The packet is sent out the network port with a single tag (SVLAN).



Note. VLAN Translation is applied when translate mode is configured using the **ethernet-service sap profile** command

Traffic Engineering and Translation at UNI and NNI Ports

This section provides important details on Traffic Engineering and Translation at UNI and NNI Ports.

Traffic Engineering at UNI Ports

- Layer 2 control frames received on UNI ports can have any TPID and are forwarded to the NNI ports with the appropriate CVLAN to SVLAN translation when required.
- In **preserve** mode, a UNI port recognizes CVLAN tag with TPID 0x8100, 0x88a8 and 0x9100. Frames with other TPIDs are considered as untagged CVLAN frames.
- Tunneling and Mac tunneling is supported for up to 7 VLAN tag headers in Layer-2 frames at a UNI port in **preserve** mode.
- In **translate** mode, the UNI port recognizes only the CVLAN tag with TPID 0x8100. Frames with other ether types are considered as untagged CVLAN frames.
- Tunneling and Mac tunneling is supported for up to 8 VLAN tag headers in Layer-2 frames at a UNI port in **translate** mode. The outermost VLAN UNI TPID is replaced by 0x8100.

Traffic Engineering at NNI Ports

- Layer 2 control frames egressing from NNI port will have the ether type value equivalent to the value configured at the NNI port (By default 0x8100) in **preserve** mode.
- NNI ports accept the frames only with ether type value configured at NNI ingress port in **preserve** mode.
- Ethernet frames received on NNI port are always forwarded to any UNI port with translated CVLAN and ethertype 0x8100, in **translate** mode.
- Tunneling and mac tunneling is supported for up to 8 VLAN tag headers at an NNI port.
- Ethernet frames received on NNI port are always forwarded to any UNI port with translated CVLAN and ethertype 0x8100, in **translate** mode.

The information on traffic engineering applied, maximum VLAN tags processed for Layer 2 control frames according to preserve or translate mode configuration for UNI and NNI ports are mentioned in the following table:

| L2 Control Frames | Mode | UNI Port Treatment | Maximum VLAN tag headers processed | Action at Egress NNI |
|-------------------|----------|--------------------|------------------------------------|----------------------|
| STP BPDU | Preserve | Tunnel | >8 | Tunnel |
| STP BPDU | Preserve | Mac Tunnel | 7 | Tunnel |
| STP BPDU | Preserve | Peer | 7 | - |
| STP BPDU | Preserve | Discard/Drop | - | - |

| L2 Control Frames | Mode | UNI Port Treatment | Maximum VLAN tag headers processed | Action at Egress NNI |
|--------------------------|-------------|---------------------------|---|-----------------------------|
| STP BPDU | Translate | Tunnel | >8 | Tunnel |
| STP BPDU | Translate | Mac Tunnel | 8 | Tunnel |
| STP BPDU | Translate | Peer | 8 | - |
| STP BPDU | Translate | Discard/Drop | - | - |
| LACP PDU | Preserve | Tunnel | 7 | Tunnel |
| LACP PDU | Preserve | Mac Tunnel | 7 | Tunnel |
| LACP PDU | Preserve | Peer | 7 | - |
| LACP PDU | Preserve | Discard/Drop | - | - |
| LACP PDU | Translate | Tunnel | 8 | Tunnel |
| LACP PDU | Translate | Mac Tunnel | 8 | Tunnel |
| LACP PDU | Translate | Peer | 8 | - |
| LACP PDU | Translate | Discard/Drop | - | - |

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 carry through the provider traffic.
- **Service Access Point (SAP)**—An SAP is associated with a VLAN Stacking service name and an 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**—An SAP profile is associated with an 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 10-14](#) for more information.

Interaction With Other Features

This section contains important information about VLAN Stacking interaction with other features enabled on OmniSwitch. 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 is similar to processing of Spanning Tree frames.
- The VLAN Stacking provider edge (PE) switch does 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 VLAN (IPMV) application has the following interactions with VLAN Stacking functionality and commands:

- IPMV operates in one of two modes:
 - Enterprise Mode.
 - VLAN Stacking Mode

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 AOS Release 6 CLI Reference Guide* for more information on VLAN Stacking commands.

Link Aggregation

- Both static and dynamic link aggregation are supported with VLAN Stacking.
- 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.
- Transparent bridging is not supported for link aggregate NNIs.

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 supported only 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. 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 10-25](#) for information on configuring VLAN Stacking to tunnel or discard Spanning Tree BPDU.
- See [“Configuring VLAN Stacking Network Ports” on page 10-18](#) for information on configuring VLAN Stacking interoperability with *legacy* Spanning Tree BPDU systems.
- A back door link configuration is not supported. Back door link occurs when there is a link between two-customer sites connected to a VLAN Stacking provider edge switch.
- A dual home configuration is not supported. Dual home 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 an 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 AOS Release 6 CLI Reference Guide* for information on the fields in the show command.

Configuring VLAN Stacking Services

Configuring a VLAN Stacking Ethernet service requires various 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 10-12](#).

- 1 Create an SVLAN.** An SVLAN is associated to a VLAN Stacking service to carry customer or provider traffic. In addition, if SVLAN is configured as an IP multicast VLAN, it can also distribute IP multicast traffic (IPMVLAN). See [“Configuring SVLANs” on page 10-16](#).
- 2 Create a VLAN Stacking service.** A service name is associated with an SVLAN to identify the customer traffic that the SVLAN carries through the provider network. See [“Configuring a VLAN Stacking Service” on page 10-17](#).
- 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 10-18](#).
- 4 Configure a VLAN Stacking service access point (SAP).** 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 10-20](#).
- 5 Configure User Network Interface (UNI) ports.** UNI ports are associated with an SAP to identify to the service from which the switch ports receive customer traffic. The SAP tunnels this traffic through the provider network. When a UNI port is associated with an SAP, the SAP parameter attributes are applied to traffic received on the UNI port. See [“Configuring VLAN Stacking User Ports” on page 10-21](#).
- 6 Associate CVLAN traffic with an SAP.** This step specifies the type of customer traffic that is allowed on UNI ports and then tunneled through the SVLAN. The type of customer traffic is associated with an SAP and applies to all UNI ports associated with the same SAP. See [“Configuring the Type of Customer Traffic to Tunnel” on page 10-22](#).
- 7 Define SAP profile attributes.** An SAP profile contains traffic engineering attributes to specify bandwidth sharing, rate limiting, CVLAN translation or double-tagging, and priority bit mapping. A default profile is automatically associated with an SAP at the time the SAP is created. If the default profile values are not sufficient, configure an SAP profile to specify different attribute values. See [“Configuring a Service Access Point Profile” on page 10-23](#).
- 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. If the default profile values are not sufficient, configure a UNI profile to specify different attribute values. See [“Configuring a UNI Profile” on page 10-25](#).

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 associating 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 the SAP UNI ports. |
| ethernet-service sap-profile | Configuring traffic engineering attributes for customer traffic that is accepted on the SAP UNI ports. |
| ethernet-service sap sap-profile | Associating a VLAN Stacking SAP with a profile. |
| ethernet-service uni-profile | Configuring how protocol control frames are processed on VLAN Stacking UNI ports. |
| ethernet-service uni uni-profile | Associating a VLAN Stacking UNI port with a profile. |

Configuring SVLANs

There are three types of SVLAN:

- **Customer SVLAN:** An SVLAN that carries customer traffic
- **Management SVLAN:** An SVLAN that carries provider management traffic
- **IPMVLAN:** An SVLAN that carries IP Multicast VLAN traffic.

SVLANs cannot be configured or modified using standard VLAN commands. As an exception, 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 **impv** (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 the status values and the name. These parameters are used to change the 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
```

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 processed by the service. The ID specified with this command identifies the SVLAN carrying traffic for the service. Each service is associated with only one SVLAN, but an SVLAN can 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 exist in the switch configuration; entering a standard VLAN ID is not allowed. See [“Configuring SVLANs” on page 10-16](#) for more information.

Once the VLAN Stacking service is created, the service 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
```

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 associated SAP before deleting 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. The 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. The port is now 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
```

When the last SVLAN association for the port is deleted, the port automatically reverts 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 legacy GVRP BPDU are tunneled on the NNI port. GVRP BPDU are dropped by default.
- **stp legacy-bpdu**—Specifies whether 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. This configuration allows 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 does 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 can 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, ensure 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 (that is, 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.

Hybrid NNI mode with QinQ and 802.1Q VLANs on same NNI port

The tagged packets received on an NNI port, with TPID other than the one configured for a port, are treated as untagged packets. If there is a TPID mismatch, these packets are accepted and flooded in the default VLAN (other than 4095) of the NNI port.

All features/properties supported with the standard VLANs (standard VLAN being configured on fixed ports) are now supported on standard VLANs, configured on NNI interface. Some of these are - mobile-tag enable/disable, mac or mac-range rule, IP-rules and so on.

The following commands are now supported on an NNI interface:

- **vlan port default**
- **vlan 802.1q**

If the default VLAN is removed from the NNI interface, then the default VLAN must be changed to 4095. It is not possible to configure VLAN 1 as default VLAN of an NNI interface.

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 10-21.
- **Customer VLANs (CVLANs).** See “Configuring the Type of Customer Traffic to Tunnel” on page 10-22.
- **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 10-23.

The VLAN Stacking components are all configured separately using different VLAN Stacking commands. The **ethernet-service sap** command is for creating an 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
```

When the SAP is deleted, all the UNI ports, CVLAN, and profile associations are automatically dropped.

A VLAN Stacking SAP identifies the following information:

- Location where customer traffic enters the provider network edge,
- The type of customer traffic to service,
- Parameters to apply to the traffic,
- The service that processes the traffic for tunneling through the provider network.

Consider the following when configuring a VLAN Stacking SAP:

- An SAP can be 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 |
| CVLAN tag | preserve (double-tag) |
| Priority mapping | fixed 0 |

The default attribute values of the profile 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 profile and associate it with the SAP. See [“Configuring a Service Access Point Profile” on page 10-23](#). Each time a profile is assigned to an 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:

```
-> no ethernet-service sap 20 uni 1/1
```

When the last SAP association for the port is deleted, the port automatically reverts to a conventional switchport, 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 10-22](#).
- 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 profile and associate it with the UNI port. See [“Configuring a UNI Profile” on page 10-25](#). Each time a profile is assigned to a UNI, the existing profile is overwritten with the new one.

Use the **show ethernet-service nni l2pt-statistics** 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 command identifies the type of customer traffic received on the SAP UNI ports that the service processes and tunnels 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
```

When the last customer traffic association is deleted from an SAP, the SAP itself is not automatically deleted. No traffic is accepted or processed by an 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 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. |
| 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 10-20](#)). If the default profile values are not sufficient, create a profile to specify different attribute values.

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.

Configuring SAP profile for Best Effort Service

In order to provision a best effort service that contains only yellow traffic service frames, the ingress bandwidth service associated to a SAP must have a CIR/CBS of 0 and a PIR/PBS value must be defined. The CIR/CBS can be set to 0 and saved in the running configuration.

Note. When CIR and CBS is set to 0 and a PIR/PBS value is defined, then it is saved in the running configuration. Use the **show configuration snapshot** command to verify whether the SAP profile configuration is saved in the running configuration.

The **ethernet-service sap-profile** command is used for this purpose.

If only CIR=0 is entered using the following command,

```
-> ethernet-service sap-profile P1 shared cir 0
```

then the associated ingress configuration is removed.

If CIR=0 and CBS = 0 is entered with the other information (PIR, PBS), the configuration is saved in the running configuration and the Tri-Color Marking (TCM) policer is set according to the entered parameters.

Ideally the PIR and PBS values must be double the CIR and CBS values. For CBS, the value can be calculated as below:

Burst-Calculation (CBS In Bits) = CIR * tc (tc = 125 msec) (For every 1/8th second committed burst packets are sent out)

Committed Burst (CBS) must be in bytes = (CIR * 125msec) bits / (8) (1 byte = 8 bits)

For example, if CIR = 20Mbps ; CBS value can be calculated as below:

CBS (in bits) = (20000000 * 125) / 1000 ==> 2500000 (is in Bits)

CBS (in Bytes) = 2500000 / 8 ==> 312 KB; PIR = 40K; PBS = 624K

```
-> ethernet-service sap-profile P1 shared cir 20k cbs 312K pir 40K pbs 624K
```

The PIR, PBS values can also be set to 0. If all the values - **cir**, **cbs**, **pir**, and **pbs** are set to zero, this configuration is not saved in the **sap-profile** running configuration. The ingress bandwidth is shared equally between the traffic ingressing through the ports associated with the SAP profile. For example,

```
-> ethernet-service sap-profile P1 shared cir 0k cbs 0K pir 0k pbs 0K
```

To delete an 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 P1
```

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, associate the profile with a VLAN Stacking SAP. When VLAN Stacking SAP is associated to the profile, the current profile associated with an 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
```

Consider 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 must 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 l2pt-statistics** 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 does 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 that 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

Create a UNI profile and 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 nni l2pt-statistics** command to display the profile associations for each UNI port.

Configuring Custom L2 Protocol

Custom L2 protocol is configured globally. The **ethernet-service custom-L2-protocol** command is used to configure a custom L2 protocol entry. For example, the following command creates a custom L2 protocol with the name p1 and MAC address 01:80:c2:00:11:11 associated to the custom-L2-protocol:

```
-> ethernet-service custom-L2-protocol p1 mac 01:80:c2:00:11:11
```

The custom L2 protocol can be applied to specific actions (tunnel, MAC-tunnel and discard). The following table describes the actions that can be associated:

| Action | Description |
|-------------------|--|
| Tunnel | Tunnels the specified PDU across the provider network without modifying the MAC address. |
| MAC-tunnel | Changes the destination MAC address to the configured tunnel MAC address of the UNI profile before forwarding. |
| Discard | Discards the specified PDU. |

Based on the configuration the custom L2 protocols are classified as qualified L2 protocols and unqualified L2 protocols.

The qualified L2 protocols are the custom L2 protocols that are fully defined with an Ether-Type and optionally a Sub-Type or ssap/dsap. The action can be set to "Tunnel", "Discard" or "Mac-Tunnel".

The unqualified L2 protocols are the custom L2 protocols that are only defined with a Mac-address or Mac-address with mask. The action can be set to "Tunnel" or "Discard".

The custom L2 protocol is associated to a UNI profile for specific packet control (Tunnel, MAC-tunnel and Discard) for proprietary protocol with multicast MAC addresses. To associate a UNI profile to a specific action use the **ethernet-service uni-profile custom-L2-protocol** command. For example, the following command specifies the action “mac-tunnel” to the custom L2 protocol “tunnel-mac-ethertype” associated to the UNI profile “profile1”:

```
-> ethernet-service uni-profile profile1 custom-L2-protocol
tunnel-mac-ethertype mac-tunnel
```

Use the **show ethernet-service sap** command to view the configuration information of the custom-L2-protocol entry.

Control Protocol Tunneling Frame Statistics

The statistics of tunneling protocols can be viewed using CLI commands. Only port level statistics can be collected in software. The statistics provided are as follows:

- **RX frame statistics at UNI port level:** On per port and per protocol basis, this is the number of frames that are trapped to CPU, number of frames tunneled, dropped, peered and MAC tunneled by the CPU operation; the source MAC of the last received frame on each port for each protocol.
- **TX frame statistics at UNI port level:** On per port and per protocol basis, this is the number of frames that are de-tunneled, and transmitted on UNI port.
- **RX frame statistics at NNI port level:** This is the number of frames that are trapped to CPU per NNI port, as their destination MAC matches with the configured tunnel MAC address and the number of frames discarded from the trapped frames.
- **RX frame statistics at UNI profile level:** This is the number of frames received on all ports bind to a UNI profile per protocol per UNI profile.

Following subsections describe the statistic types:

UNI port Statistics: The UNI port statistics are collected at software level, that is, statistics of frames trapped to CPU. The frames are processed by software if frames' action is MAC-tunnel or Peer and/or the frame cannot be classified in the hardware. Also, the frames received on NNI are processed by software if they need to be de-tunneled. At this level, the following statistics are collected per port per protocol.

- The number of UNI frames received on UNI ports trapped to CPU.
- The number of UNI frames trapped to CPU for each action such as drop/peer/tunnel-mac/tunnel.
- The number of de-tunneled frames before they are flooded on UNI port.
- The source MAC of the last received frame per protocol per port.

NNI Port Statistics: These statistics provide the details of number of frames trapped on an NNI per port and the number of frames discarded. The number of frames received on NNI ports and trapped to the CPU.

The number of frames discarded, this number is the count of frames that do not match any de-tunnel criteria for the NNI frames that are trapped to CPU.

UNI Profile Statistics: The UNI profile level statistics are collected in hardware, for DROP and TUNNEL and software for TRAP. Per UNI profile statistics are statistics of all the frames received per protocol on all the ports bind to a UNI profile. Since these are collected in HW and a protocol in HW is represented by its IPCL entry, there are cases in which one IPCL entry may represent more than one protocol. In such cases the statistics displayed represents all the frames that match with the corresponding

IPCL entry. Along with the number of frames received per protocol, the action taken, that is, the hardware action configured for the IPCL entry is also shown.

Also, the total number of frames received per UNI profile is also displayed.

In the case of UNI profile, following information is provided:

- Total number of frames received per UNI profile.
- Total number of frames received per protocol, only per IPCL, per UNI profile for hardware treatment DROP and TUNNEL. For L2 protocols, which have hardware treatment set as TRAP, UNI-profile statistics will account the software statistics.
- The actual treatment applied in hardware for each L2 protocol.

For more information on the statistic commands see [“VLAN Stacking Commands” on page 7-1](#)

Control HW Tunneling

A new feature is introduced to unconditionally forward all MAC tunnel packets irrespective of its UNI profile as and when required.

This feature can be enabled or disabled using global flag "**noMacTunnelFeature**" in the *AlcatelDebug.cfg* file. The feature is functional after reboot. The global tunnel-mac PCL entry is not applied if this flag is enabled. Therefore, all the tunneled-mac PDUs will be hardware tunneled from NNI.

The flag "noMacTunnelFeature" is set to enable in *AlcatelDebug.cfg* file using following:

- Debug set noMacTunnelFeature 1
- Reboot the switch after setting the variable in debug file, as the tunnel-mac PCL entry gets applied during bootup.

Once the system gets rebooted, all the mac-tunnel packets will not be trapped to CPU at NNI and will get tunneled from hardware.

Warning: When the "noMacTunnelFeature" is enabled, all MAC tunnel packets are tunneled and there is no check on which protocol it is associated with.

Configuring MAC-Tunneling for SVLAN

When MAC-tunneling is enabled globally, the Generic Bridge PDU Tunneling (GBPT) frames of the UNI profile which do not have MAC-tunneling configured are also captured to the CPU and transferred at CPU rate at the NNI.

In order to avoid the GBPT packets from being rate limited to CPU at the NNI, MAC-tunneling can be enabled on per SVLAN for a UNI profile. This allows the GBPT packets to be tunneled through hardware at wire rate and the MAC-tunneled packets are trapped to CPU on a per SVLAN basis. To enable MAC-tunneling on per SVLAN basis, MAC-tunneling has to be disabled globally.

The following sections describe the various configurations required to activate this functionality.

Global MAC-Tunneling Status

On enabling the MAC-tunneling globally the GBPT frames are trapped to CPU at the NNI. To enable or disable the MAC-tunneling status globally use the **ethernet-service mac-tunneling** command. For example:

```
-> ethernet-service mac-tunneling enable
-> ethernet-service mac-tunneling disable
```

Note. When MAC-tunneling is enabled globally, per SVLAN MAC-tunneling configuration will not be active.

When MAC-tunneling is disabled globally, the MAC-tunnel status of the SVLANs configured will be active.

SVLAN MAC-Tunneling Configuration

The MAC-tunneling can be enabled on per SVLAN for a UNI profile. This allows the GBPT packets to be tunneled through hardware at wire rate and the MAC-tunneled packets are trapped to CPU on a per SVLAN basis. To configure the MAC-tunneling on a per SVLAN basis, use the **ethernet-service svlan mac-tunneling** command. For example:

```
-> ethernet-service svlan 1000 mac-tunneling enable name "VLAN 1000"
-> ethernet-service svlan 1000 mac-tunneling disable name "VLAN 1000"
```

Note. Maximum four SVLAN can have MAC-tunnel enabled simultaneously.

MAC-tunneling must be disabled globally before MAC-tunneling is enabled on per SVLAN.

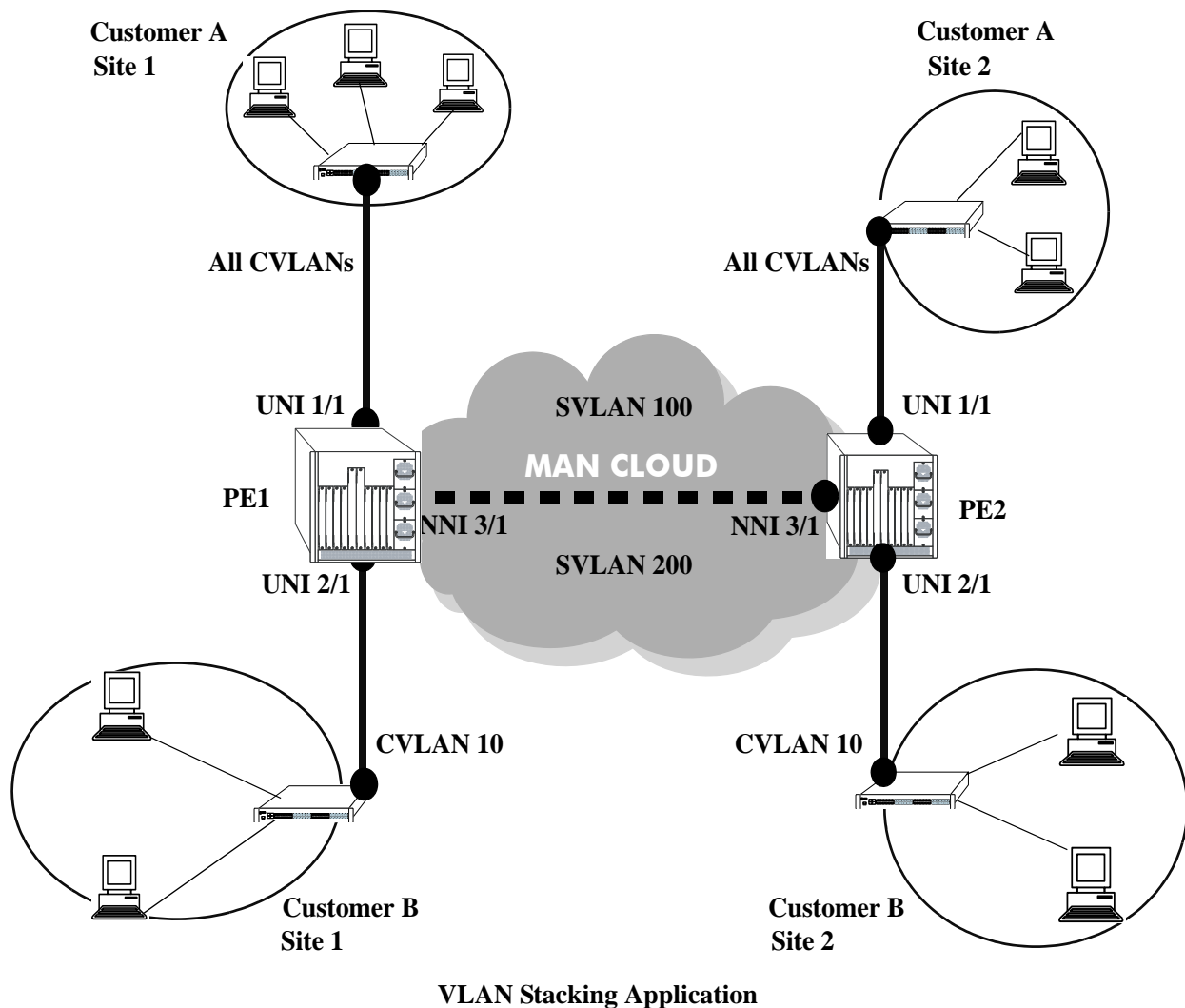
To view the status of the MAC-tunnel configured for the SVLAN, use the **show vlan** command with the VLAN ID parameter.

VLAN Stacking Application Examples

The VLAN Stacking feature provides the ability to connect multiple customer sites transparently over a single shared service provider network. This section provides 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 following illustration shows the sample VLAN Stacking configuration. In this configuration, the provider edge bridges encapsulates 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 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 10-30](#), 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 an SAP profile on PE1 *and* PE2 that maps to 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. It is not necessary to associate the profile with SAP 20 as this profile is applicable only to Customer B traffic.

```
-> 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 when 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.

Two types of loopback tests supported with this implementation:

- Inward loopback
- 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 generates 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 performs the egress or ingress loopback operation for the test.
- The type of test to run (outward or inward loopback).
- For inward loopback, it is mandatory to provide the destination MAC address and loopback port. The source MAC address and VLAN ID is optional.
- For outward loopback, it is mandatory to provide the destination MAC address, loopback port, and VLAN ID. The source MAC address is optional.
- For outward loopback test, SAP ID can be configured. SAP ID is required to fetch the mode (translate or preserve) from the input SAP ID. Since multiple SAPs can be associated with the same UNI Port, SAP ID is used to uniquely identify the SAP. If the SAP ID is not specified as an option in the outward loopback test, then SAP with the lowest SAP ID is used.

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 destination-mac 00:00:00:cc:aa:bb loopback-port
1/1 source-mac 00:00:00:dd:aa:01 vlan 1001 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 swaps 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 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

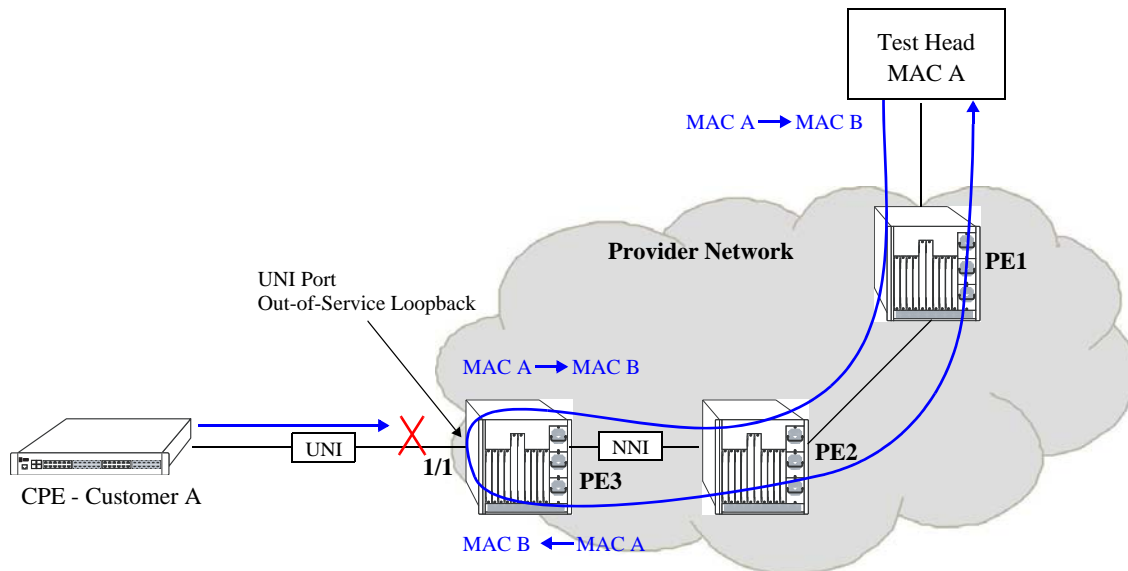
An outward loopback test loops back test frames egressing on a specific port. The source and destination MAC addresses of the test frames are swapped and the frames are redirected back to the port on which they were initially received and learned (the redirect port). The redirect port is not configured as a part of the test profile; the source learning function determines the port to be used 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 can cause network reachability problems.



Outward (Egress) Loopback Test Example

In this outward loopback test example:

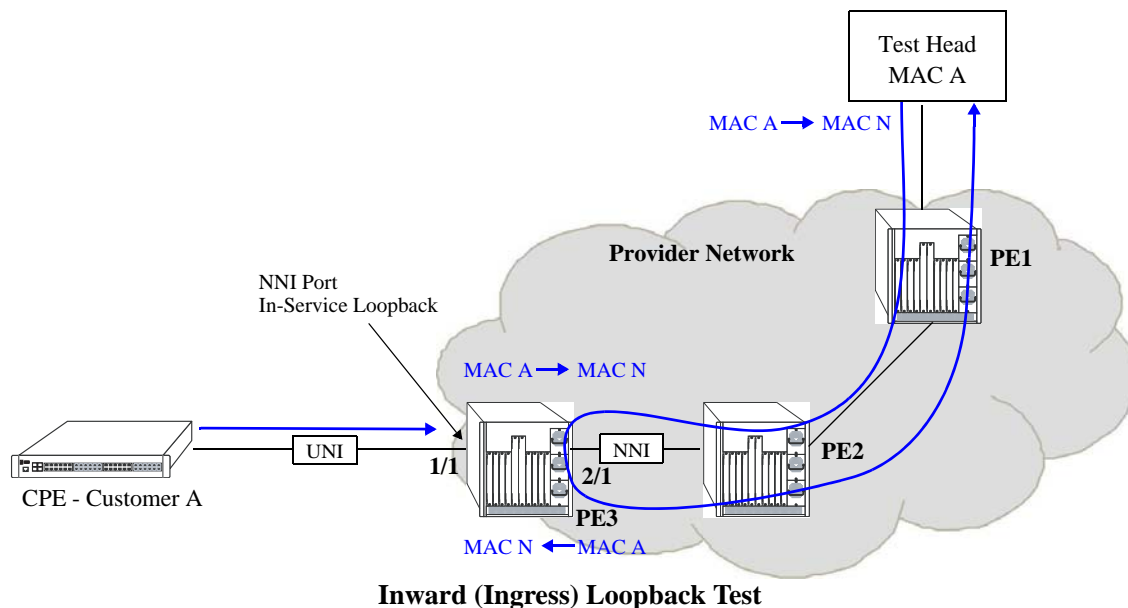
- An outward loopback test profile is configured and enabled for UNI port 1/1 on PE3. The source MAC address for the profile is that of the test head (MAC A); the destination MAC address is a static MAC address configured for the UNI port (MAC B).
- UNI port 1/1 on PE3 is out of service for customer 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 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 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 an 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:

| | |
|---|--|
| ethernet-service uni-profile custom-L2-protocol | 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 nni l2pt-statistics | Displays profile associations for UNI ports. |
| show ethernet-service uni l2pt-statistics | 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. |
| show ethernet-service sap | Displays configuration information of the specific custom-L2-protocol entry if specified or displays information of all the configured custom-L2-protocol entries in the system. |
| ethernet-service uni-profile | Creates a UNI profile that is used to specify how to process control packets ingressing on UNI ports. |
| ethernet-service uni-profile custom-L2-protocol | Associates a custom-L2-protocol entry to a UNI profile. |
| ethernet-service uni uni-profile | Associates a VLAN Stacking UNI profile with a UNI port. |
| show ethernet-service nni l2pt-statistics | Displays the profile associations for VLAN Stacking User Network Interface (UNI) ports. |
| show ethernet-service uni-profile | Displays the statistics of all protocols configured per UNI port. |
| show ethernet-service uni-profile l2pt- statistics | Displays the profile statistics for VLAN Stacking User Network Interface (UNI) profiles. |
| clear ethernet-service uni l2pt-statistics | Clears the statistics of all protocols configured per UNI port. |
| clear ethernet-service uni-profile l2pt-statistics | Clears the statistics of all UNI profiles. |
| clear ethernet-service nni l2pt-statistics | Clears all Network Network Interface (NNI) ports statistics. |

For more information on the resulting displays from the show commands, see the *OmniSwitch AOS Release 6 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 10-12.

11 Using 802.1Q 2005 Multiple Spanning Tree

The Alcatel 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 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 AOS Release 6 CLI Reference Guide*. For more information about Spanning Tree configuration commands as they apply to all supported protocols (STP, RSTP, and MSTP), see [Chapter 12, “Configuring Spanning Tree.”](#)

The following topics are included in this chapter as they relate to the Alcatel implementation of the MSTP standard:

- [“MST General Overview” on page 11-4.](#)
- [“MST Configuration Overview” on page 11-9.](#)
- [“Using Spanning Tree Configuration Commands” on page 11-10.](#)
- [“MST Interoperability and Migration” on page 11-11.](#)
- [“Quick Steps for Configuring an MST Region” on page 11-13.](#)
- [“Quick Steps for Configuring MSTIs” on page 11-15.](#)
- [“Verifying the MST Configuration” on page 11-18.](#)

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 6350, 6450 |
| 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 AOS Release 6 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 11-4.](#)
- [“Comparing MSTP with STP and RSTP” on page 11-7.](#)
- [“What is a Multiple Spanning Tree Instance \(MSTI\)” on page 11-7.](#)
- [“What is a Multiple Spanning Tree Region” on page 11-8.](#)
- [“What is the Internal Spanning Tree \(IST\) Instance” on page 11-9.](#)
- [“What is the Common and Internal Spanning Tree Instance” on page 11-9.](#)

How MSTP Works

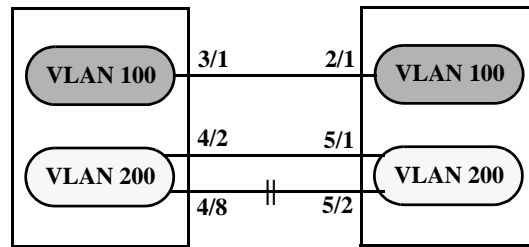
MSTP, as defined in the IEEE 802.1Q 2005 standard, is an enhancement to the IEEE 802.1Q Common Spanning Tree (CST). The CST is a single spanning tree that uses 802.1D (STP) or 802.1w (RSTP) to provide a loop-free network topology.

The Alcatel flat spanning tree mode applies a single CST instance on a per switch basis. The 1x1 mode is an Alcatel 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 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.



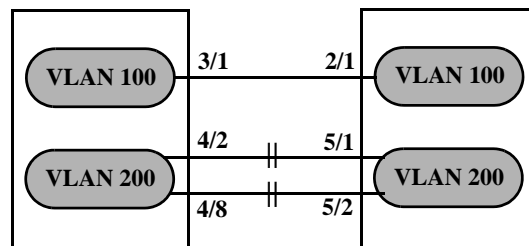
1x1 Mode STP/RSTP

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.

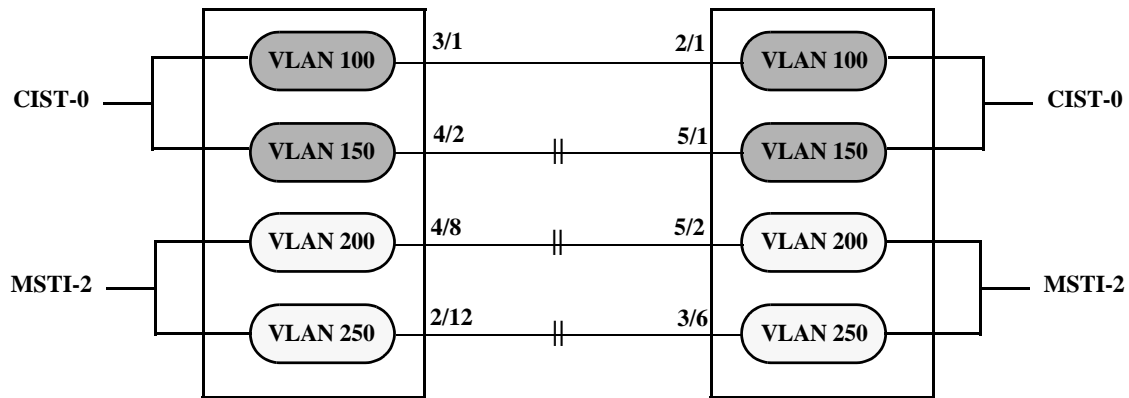
- The connections between 4/8 and 5/2 and 4/2 and 5/1 are seen as redundant because they are both controlled by the VLAN 200 Spanning Tree instance and connect to the same switches. The VLAN 200 Spanning Tree instance determines which connection provides the best data path and transitions the other connection 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 contains only MSTI information.

See [“Quick Steps for Configuring MSTIs” on page 11-15](#) 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 11-7](#) 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 11-9](#) 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 11-8](#) 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 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 11-9](#) for more information.

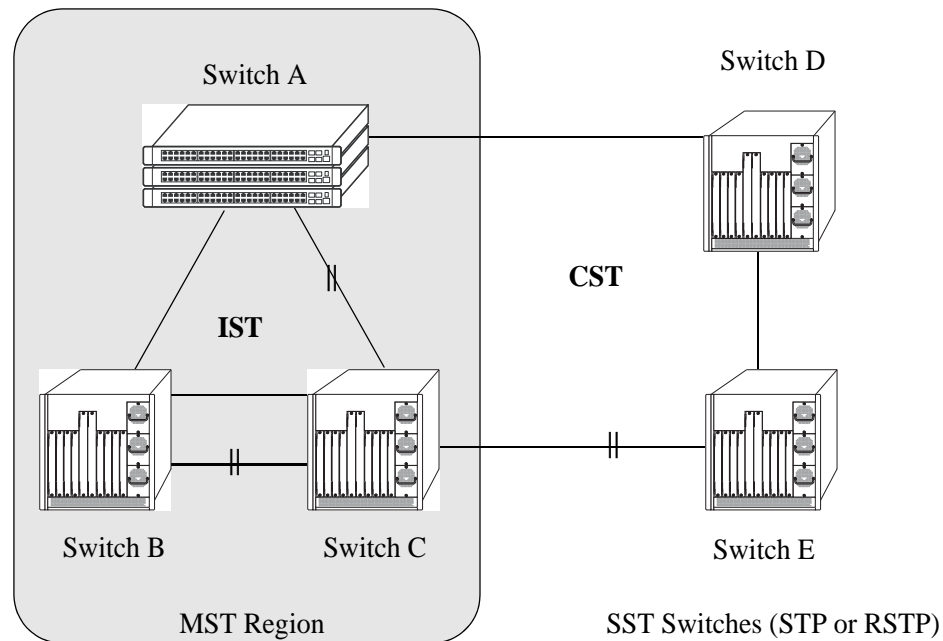
What is a Multiple Spanning Tree Region

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 11-9](#) 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 11-13](#) 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 11-10](#) 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 11-11](#) 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 11-4](#) 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 11-8](#) 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 11-7 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 determines 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 11-13 and “Quick Steps for Configuring MSTIs” on page 11-15.

Using Spanning Tree Configuration Commands

The Alcatel 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 (for example, **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 12, “Configuring Spanning Tree.”](#)

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 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 11-10](#) 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 12, “Configuring Spanning Tree,”](#) 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 makes 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 11-4](#) 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 11-10](#) 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 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:

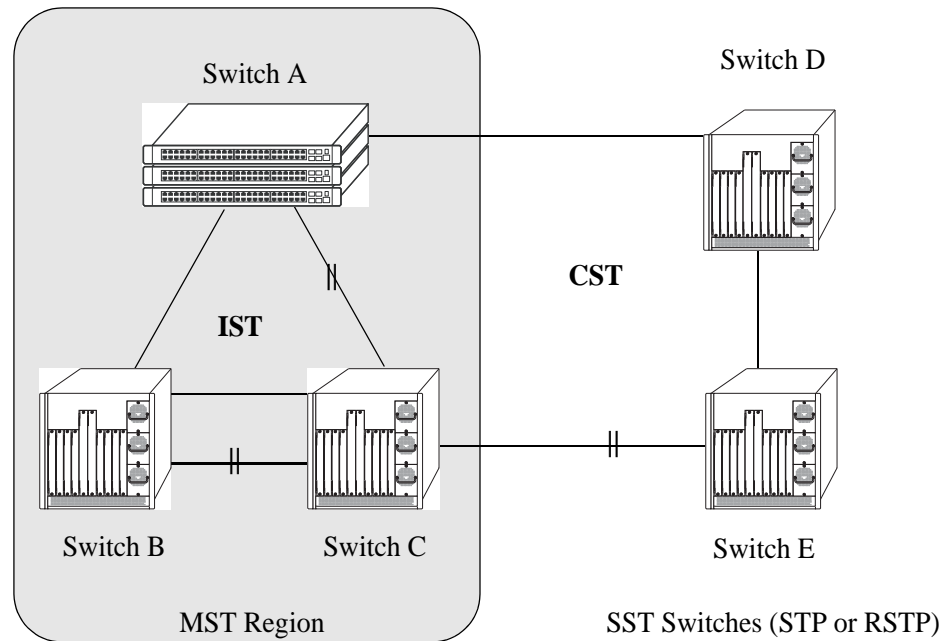
- Making a backup copy of the switch **boot.cfg** file before changing the protocol to MSTP is highly recommended. Having a backup copy makes it easier to revert to the non-MSTP configuration if necessary. Once MSTP is active, commands are written in their explicit form and not compatible with previous 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 11-4](#) 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 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 11-15](#) 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
```

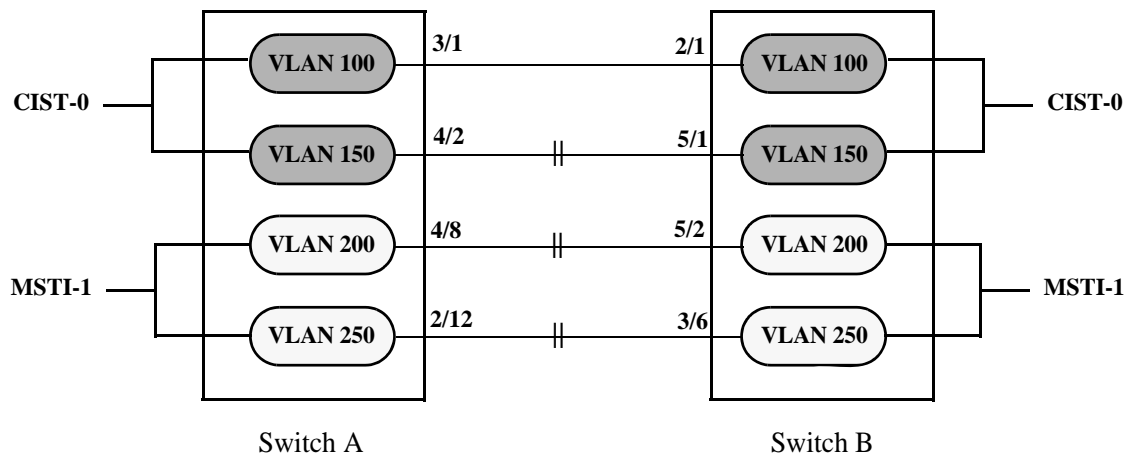
All switches configured with the exact same values as shown in the above example are considered members of the Alcatel Marketing MST region.

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 11-10](#) 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
```

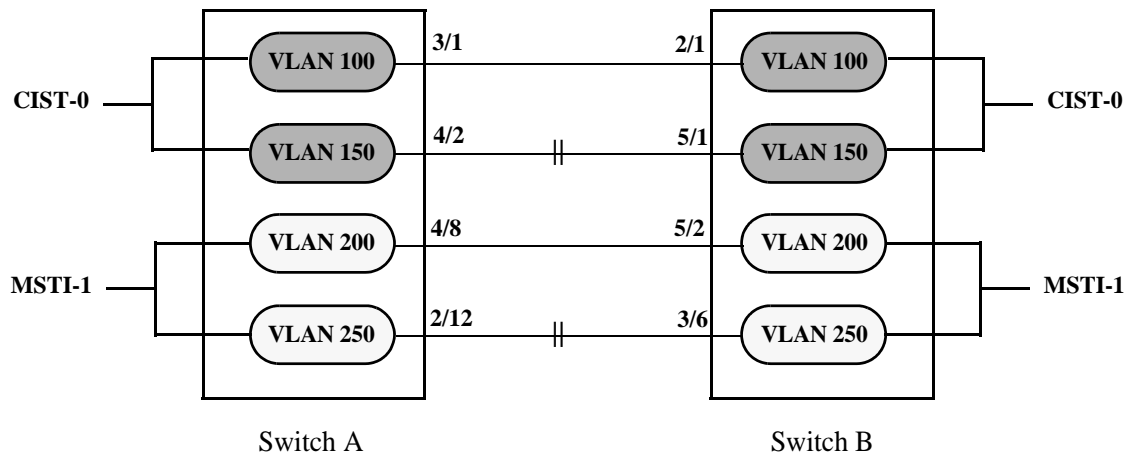
By default, all VLANs are associated with the CIST instance. As a result, VLANs 100 and 150 do not require any configuration to map them to the CIST instance.

- 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 11-15](#). 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 11-17](#).



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 contains only MSTI information. See [“How MSTP Works” on page 11-4](#) 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 AOS Release 6 CLI Reference Guide*.

12 Configuring Spanning Tree

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 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 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Selecting the switch Spanning Tree operating mode (flat or 1x1) on [page 12-15](#).
- Configuring Spanning Tree bridge parameters on [page 12-21](#).
- Configuring Spanning Tree port parameters on [page 12-30](#).
- Configuring Ring Rapid Spanning Tree on [page 12-43](#).
- Configuring an example Spanning Tree topology on [page 12-44](#).

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 6350, 6450 |
| 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 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 AOS Release 6 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 | <code>bridge rrstp</code> | Disabled |
| Number of rings | <code>bridge rrstp ring</code> | 0 |
| Ring status | <code>bridge rrstp ring</code> <code>bridge rrstp ring status</code> | Disabled |

Spanning Tree Overview

Alcatel 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 inter-operate 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 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 11, “Using 802.1Q 2005 Multiple Spanning Tree,”](#) for specific information about configuring MSTP. For more information about using RRSTP, see [“Using RRSTP” on page 12-42.](#)

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 participates 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 12-35 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 12-24 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 <i>OmniSwitch AOS Release 6350/6450 Hardware Users Guide</i> 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 12-34 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).

Topology Change Notification

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 then forwards 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 [“Restricting TCN Propagation” on page 12-41](#).

Detecting the Source of Topology Changes

The following information and logging mechanisms are available on each switch to help identify the source of topology changes within an active network:

- The port on which the last TCN was received on the local switch. The “Last TC Rcvd Port” field of the **show spantree** command displays the switch port on which the last TCN was received. The “Last TC Rcvd Port” default value is **none** and supported only for RSTP and MSTP protocols. For linkagg, it will be displayed as `<0/linkagg_num>`.
- Switch logging entries to identify root port and root bridge changes for all Spanning Tree protocols (flat STP, flat RSTP, 1x1 STP, 1x1 RSTP, MSTP). For example:

For Root Port Change:

```
MON JAN 31 06:37:41 2000 STP info Root port Change for VLAN/STP-ID 3/4099 on
port 2/21
```

For Root Bridge Change:

```
MON JAN 31 08:00:51 2000 STP info New Root Bridge Change for VLAN/STP-ID 3/4099
```

- Topology change storm detection to identify excessive topology changes for all Spanning Tree protocols (flat STP, flat RSTP, 1x1 STP, 1x1 RSTP, MSTP). The switch uses internal calculations based on the number of topology changes within a specific period of time to determine if the number of topology changes exceeds a specific threshold. When this threshold value is reached, switch logging entries are triggered as a warning of potential instability within the network.

For example:

SWLOG for storm due to number of TC received on a port:

For VLAN + RSTP:

```
MON JAN 31 06:37:41 2000 STP warning Topology Change Storm detected for VLAN 3
on PORT 2/21
```

For FLAT + CIST:

```
MON JAN 31 06:30:11 2000 STP warning Topology Change Storm detected on PORT 1/1
```

For FLAT + MSTP CIST:

```
MON JAN 31 07:27:41 2000 STP warning Topology Change Storm detected for CIST on
PORT 2/5
```

For FLAT + MSTP MSTI:

```
MON JAN 31 06:30:11 2000 STP warning Topology Change Storm detected for MSTI 1
on PORT 2/6
```

SWLOG for storm due to topology change on a port:

For VLAN + RSTP:

```
MON JAN 31 06:37:41 2000 STP warning Topology Change Storm generated for VLAN 3
on PORT 2/21
```

For FLAT + CIST:

```
MON JAN 31 06:30:11 2000 STP warning Topology Change Storm generated on PORT 1/1
```

For FLAT + MSTP CIST:

```
MON JAN 31 07:27:41 2000 STP warning Topology Change Storm generated for CIST on
PORT 2/5
```

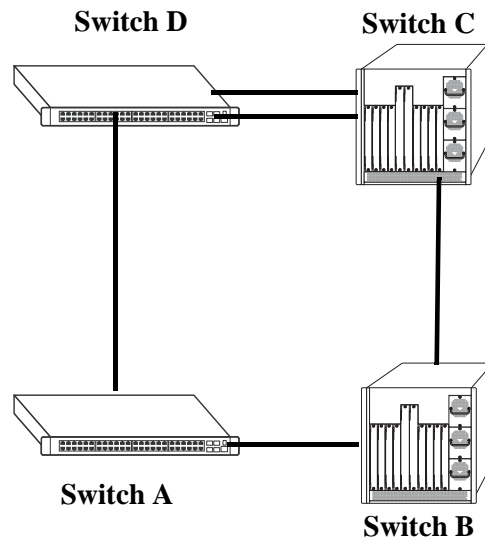
For FLAT + MSTP MSTI:

```
MON JAN 31 06:30:11 2000 STP warning Topology Change Storm generated for MSTI 1
on PORT 2/6
```

For more information about the switch logging utility, see [Chapter 34, "Using Switch Logging."](#)

Topology Examples

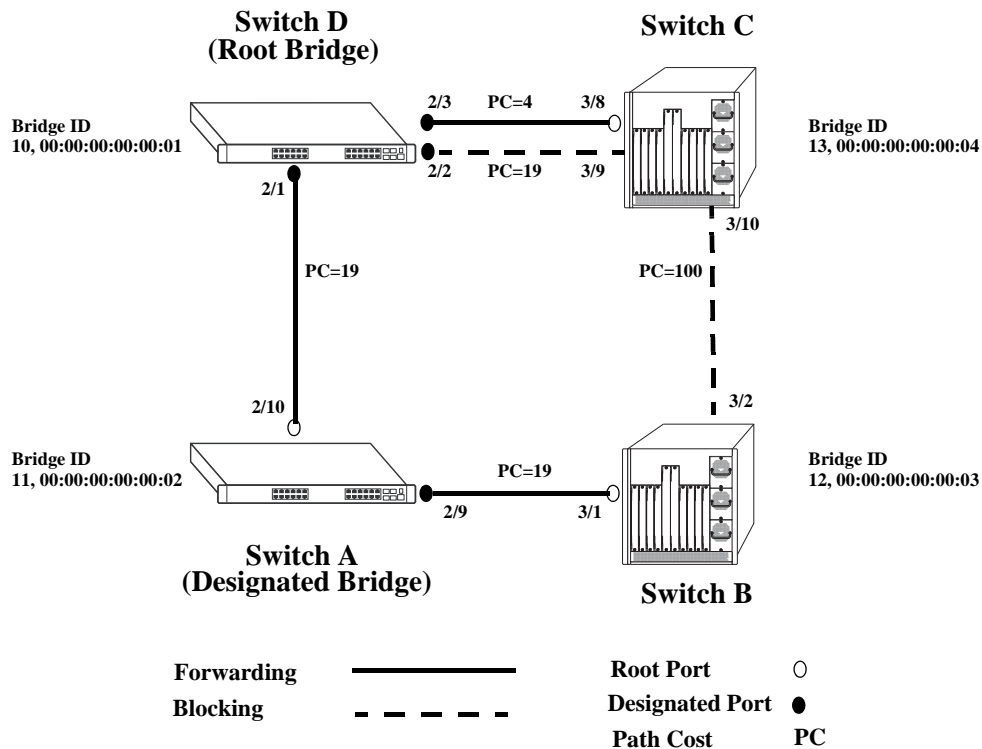
The following diagram shows an example of a physical network topology that incorporates data path redundancy to ensure fault tolerance. These redundant paths, however, create loops in the network configuration. If a device connected to Switch A sends broadcast packets, Switch A floods the packets out all of its active ports. The switches connected to Switch A in turn floods the broadcast packets out their active ports, and Switch A eventually receives the same packets back and the cycle starts 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 transitions 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 rrstp ring vlan-tag** command. For more information about this command, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Using Flat Spanning Tree Mode

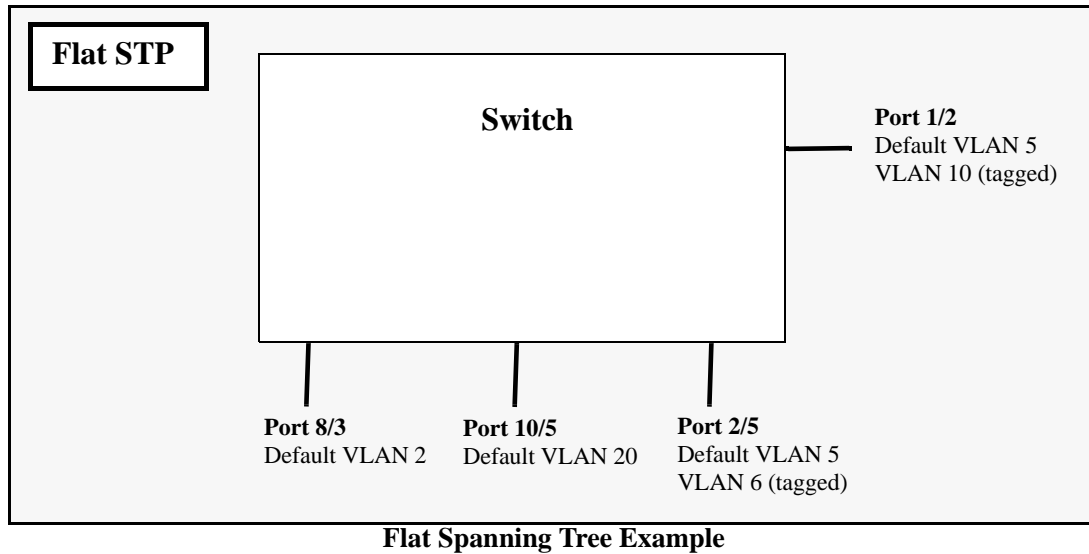
Before selecting the flat Spanning Tree mode, consider the following:

- 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 11, “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

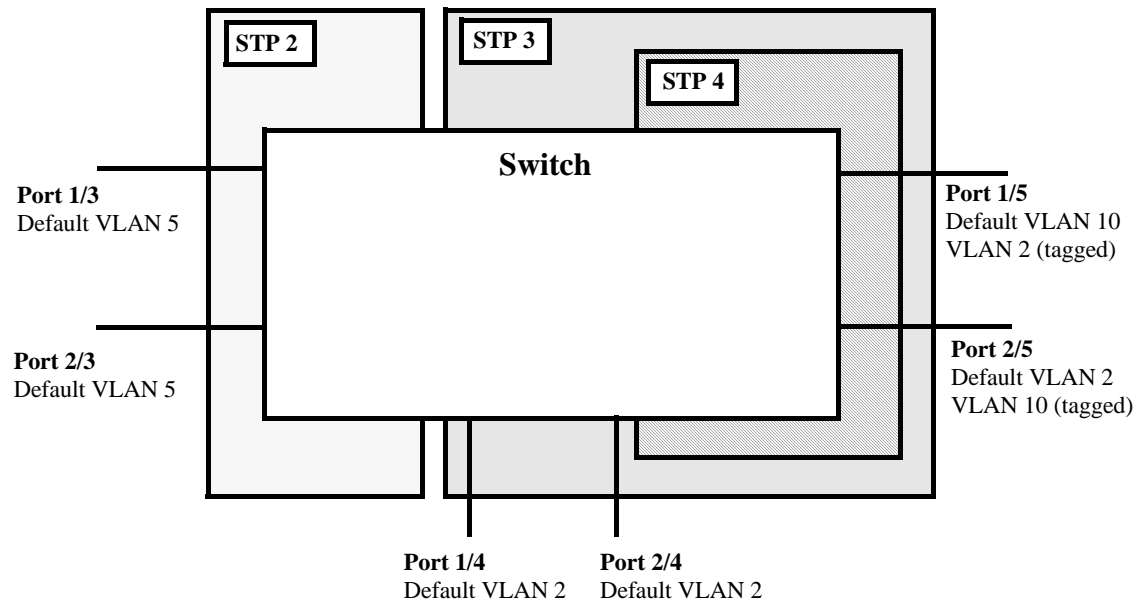
Before selecting the 1x1 Spanning Tree operating mode, consider the following:

- 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 has 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 does not block one of them. However, if two ports within VLAN 10 both connect to the same switch, then STP transitions 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 inter-operate 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 enables 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 inter-operate 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 inter-operate 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 (that is, 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 interoperates 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 handles 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 causes 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 (that is, disable state). Once a PVST+ BPDU is received, it then handles 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 processes BPDUs as follows:

If the default VLAN of a port is VLAN 1 then:

- Send and receive IEEE untagged BPDUs for VLAN 1
- Do not 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
- Do not 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 avoids 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 results in an error message. Also, the existing 1x1 priority values is restored when changing from PVST+ mode back to 1x1 mode. For more information on priority, refer [“Configuring the Bridge Priority” on page 12-24](#).
- 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 12-24](#).
- 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 is consistent when connecting ports between OmniSwitch and Cisco Switches. For more information on path cost mode, refer [“Configuring the Path Cost Mode” on page 12-28](#).
- 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 12-37](#).

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 remains 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 remains 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 AOS Release 6 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 configures. 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 11, “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 AOS Release 6 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. |
| 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. |

| Commands | Type | Used for ... |
|-------------------------------------|------|---|
| 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 (for example, **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 serves 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 causes 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 results 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 (for example, **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 11, “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 waits 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 adopts 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 (for example, **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 attempts 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 (for example, **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 adopts 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 (for example, **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 has a higher PPC value that advertises 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:

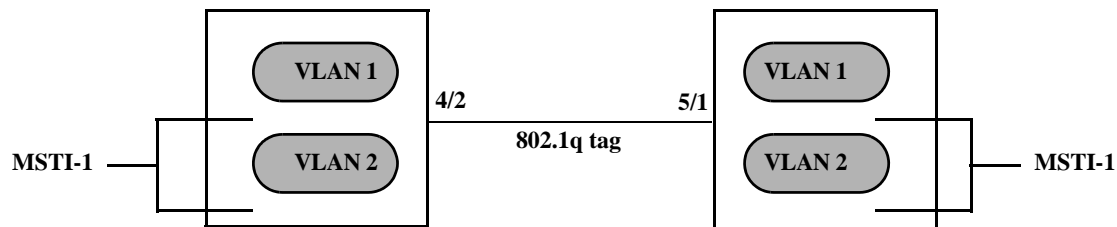
```
-> bridge path cost mode 32bit
```

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 has to be configured in the same way in all the switches, so that, the path costs for similar interface types is consistent when connecting ports between OmniSwitch and Cisco Switches.

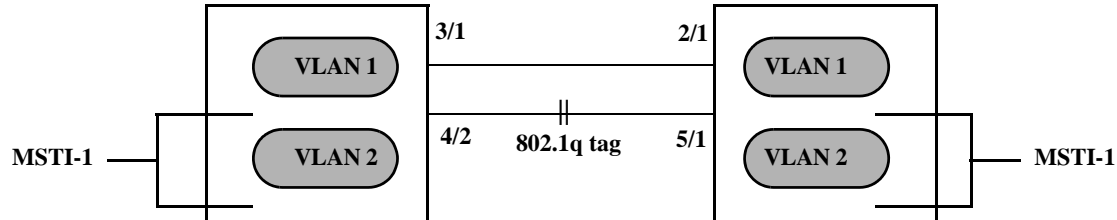
Using Automatic VLAN Containmentment

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 configures. 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 11, "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 AOS Release 6 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 (for example, **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 25, "Configuring Static Link Aggregation,"](#) and [Chapter 26, "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 (for example, **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 11, "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 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 12-28](#) 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 (for example, **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 11, "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 |
| | 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 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 operates 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 determines 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 (for example, **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 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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, operationally reverts 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 has to 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 connects to a host (PC, workstation, server, and so on.) as edge ports so that these ports transitions 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 assumes 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 (for example, **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 25, “Configuring Static Link Aggregation,”](#) and [Chapter 26, “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 is 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 txholdcount** 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 has to 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 does not occur if an RRSTP frame is lost. However, MSTP convergence still takes 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 continues without interruption.
- A single switch can participate in up to 128 RRSTP rings.

Configuring RRSTP

This section describes how to use Alcatel'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 12-43](#).
- 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 12-43](#).

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
```

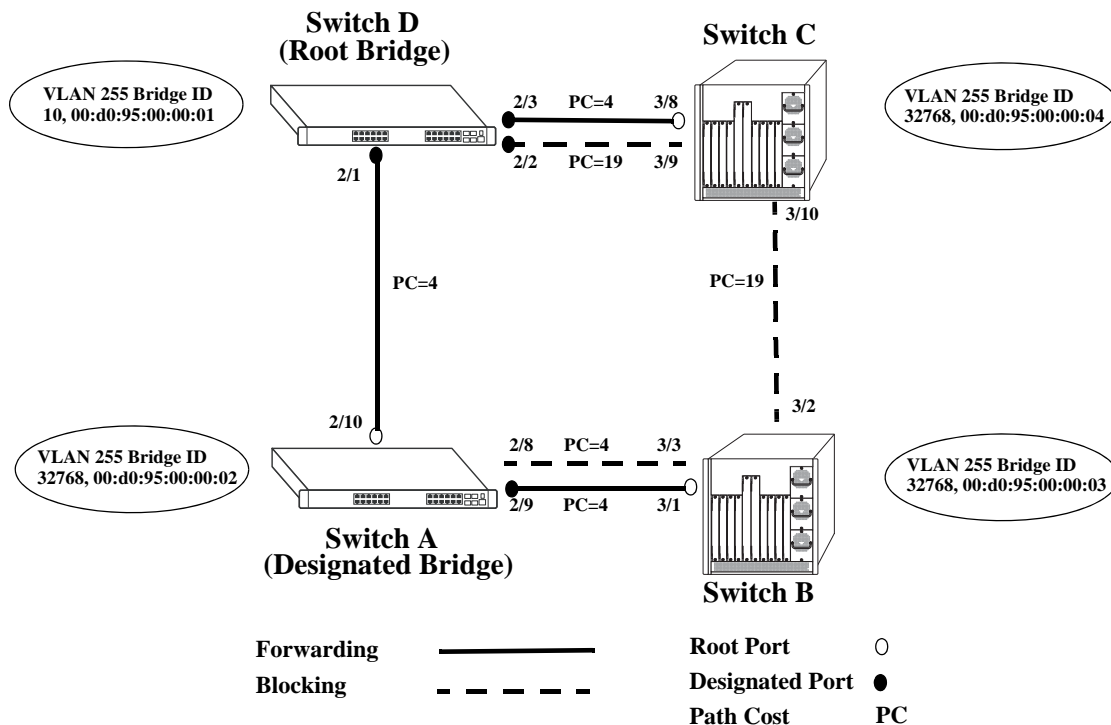
Sample Spanning Tree Configuration

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 11, “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 12-44](#).

- 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 12-44](#) 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 AOS Release 6 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 12-45.

13 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.

Alcatel-Lucent OmniSwitch supports ERIPv2 according to the ITU-T recommendation. G.8032 03/2010 in the current AOS version. The previous AOS versions support ERIPv1.

The ERIPv2 implementation helps maintain a loop-free topology in multi-ring and ladder networks that contain interconnection nodes, interconnected shared links, master rings and sub-rings.

The following chapter details the different functionalities and configuration settings required for ERIPv2.

In This Chapter

This chapter provides an overview about how ERIPv2 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 AOS Release 6 CLI Reference Guide*.

The following information and configuration procedures are included in this chapter for ERIPv2:

- [“ERIPv2 Overview” on page 13-4.](#)
- [“ERIPv2 Basic Operation” on page 13-6](#)
- [“Interaction With Other Features” on page 13-9.](#)
- [“Quick Steps for Configuring ERIPv2 with Standard VLANs” on page 13-10](#)
- [“Quick Steps for Configuring ERIPv2 with VLAN Stacking” on page 13-11](#)
- [“ERIPv2 Configuration Overview and Guidelines” on page 13-12..](#)
- [“ERIPv2 Application Example” on page 13-22.](#)
- [“Verifying the ERIPv2 Configuration” on page 13-25.](#)

ERPV2 Specifications

The following table specifies the ERPv2 related specification

| | |
|---|---|
| ITU-T G.8032 03/2010 | Ethernet Ring Protection version 2 (Multi Rings and Ladder networks supported) (Hold off timer, Lockout, Signal degrade SD, RPL Replacement, Forced Switch, Manual Switch, Clear for Manual/Forced Switch, Dual end blocking not supported) |
| ITU-T Y.1344 2010 802.1ag | ERPV2 packet compliant with OAM PDU format for CCM |
| Supported Platforms | OmniSwitch 6450 Metro license required for OmniSwitch 6450 |
| Maximum number of rings per node | 8 |
| Maximum number of rings per ring port | 1 |
| Maximum number of nodes per ring | 16 (recommended) |
| Range for ring ID | 1 - 2147483647 |
| Range for remote MEP ID | 1 - 8191 |
| Range for wait-to-restore timer | 1 - 12 minutes |
| Range for guard timer | 1 - 200 centi-seconds |
| Maximum Link Failure Detection Time + Source Learning Database Flush Time | 12.6ms |
| Maximum protection switching completion time. | 50ms |
| ERPV2 multicast MAC address | 01-19-A7-00-00-01 |

ERPV2 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 |
| ERPV2 ring Virtual Channel. | erp-ring virtual-channel | Enabled |
| Revertive mode on a specified node | erp-ring revertive | Enabled |

ERPV2 Overview

The ERPv2 implementation of ITU-T G.8032 03/2010 supports multi-ring and ladder networks with interconnection nodes, interconnected shared links, master rings and sub-rings. The following features are also supported:

- R-APS Virtual Channel
- Revertive/Non-Revertive modes

A shared link can be a part of one master ring. The sub-rings connected to the interconnection nodes are open.

ERPV2 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.

RMEPID — Remote Maintenance End Point Identifier.

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 (for example, 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.

FDB — The Filtering Database that stores filtered data according to the R-APS messages received. This database also maintains an association table that identifies the master rings for a given sub-ring.

BPR — The Blocked Port Reference that identifies the ring port (0 for interconnection node or sub-ring, 1 for master ring) that is blocked. The BPR status is used in all R-APS messages.

CCM — When an Ethernet ring contains no ERP capable nodes, CCM (Continuity Check Messages) are required to monitor the ring-port connectivity across the L2 network.

MEG and MEL — The switches in the Management Entity Group with given priority as MEG level (MEL).

NR and SF — Not Reachable and Signal Failure specify the status messages that can be sent as part of the R-APS messages.

ERPV2 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 [“ERPV2 Specifications” on page 13-2](#) for timer defaults and valid ranges.

Guard Timer. When the failed link recovers, a ring node starts 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 [“ERPV2 Specifications” on page 13-2](#) for timer defaults and valid ranges..

ERPV2 Basic Operation

The enhanced ERPv2 functionality supports multi-ring and ladder networks that contain interconnection nodes, interconnected shared links, master rings and sub-rings. Multiple ERPv2 instances are supported per physical ring.

A shared link can only be part of the master ring. The sub-rings connected to the interconnection nodes are not closed and cannot use the shared links.

Consider the following OmniSwitch multi-ring and ladder network with the Master or Major Ring with five ring nodes. The sub-ring, ladder networks, RPLs and shared links are also depicted as part of the illustration.

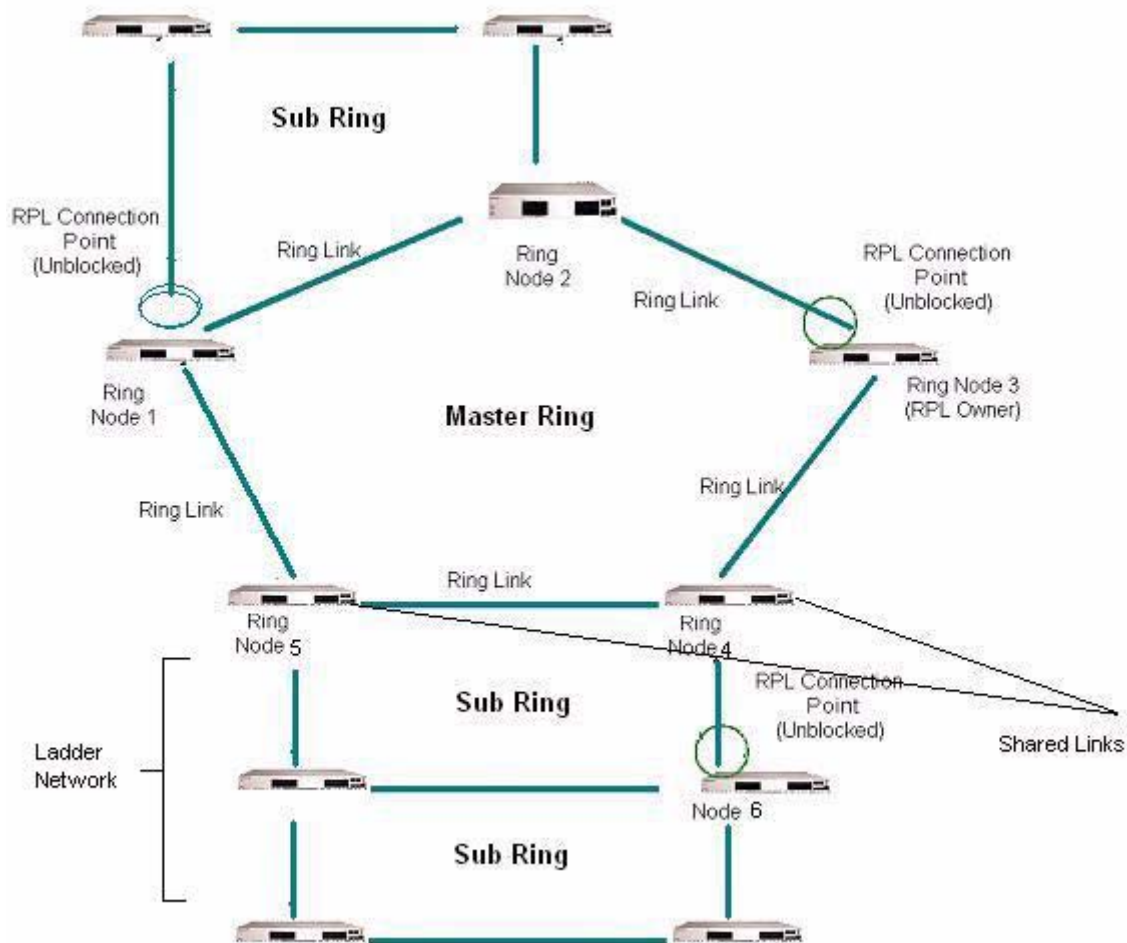


Illustration of ERPv2 on Multi Ring and Ladder Network with RPLs and Shared Links

R-APS Virtual Channel

ERPV2 supports two implementation options for R-APS control channel of the sub-ring.

- **Virtual Channel Enabled** - R-APS messages are encapsulated and transmitted over an R-APS Virtual channel configured on the major ring.
- **Virtual Channel Disabled** - R-APS messages are terminated at the interconnection nodes but not blocked at RPL of the sub-ring. RPL ports are unblocked when all nodes are active (there is no failed node).

For details on how to enable and disable R-APS virtual channel, see the section - [“Enabling and Disabling R-APS Virtual Channel” on page 13-18](#)

The R-APS channels are not shared across rings. Each ring must have its own R-APS Channel.

- The R-APS virtual channels of the sub-rings are automatically **closed** using the master ring. R-APS messages from the sub-ring on the interconnection node are forwarded as normal data to and only to the master ring ports.
- The R-APS messages use a static destination MAC address of 01-19-A7-00-00-01. R-APS messages must be tagged in order to identify the ring ID.

Note. The Service VLAN must be tagged, there is no support for "untagged" service VLAN in ERPv2. The sub-ring and master ring cannot use the same service VLAN.

Revertive / Non-Revertive Mode

Revertive mode is configured for compatibility between ERPv1 and ERPv2 nodes in the same ring. When the ERPv2 node is operating with ERP v1 node in the same ring, it operates in revertive mode regardless of user configuration.

Non-Revertive mode: Under non-revertive mode, when the failure condition recovers, the port that has been blocked stays blocked and the unblocked RPL stays unblocked.

An exclusive clear operation can also be performed for non-revertive mode and revertive mode using the ERPv2 CLI to clear any pending state. For details on CLI usage, see the section [“Configuring Revertive and Non-revertive Mode” on page 13-19](#).

ERPV2 and RRSTP Differences

ERPV2 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

STP is automatically disabled when ERP is enabled on any port.

1X1 Mode

- Every VLAN will be a part of separate STG and in that STG, ERP port will be controlled by ERP control process and others by STP.

Flat Mode

- RRSTP and ERP cannot be configured on the same port.
- STP will be administratively disabled on ERP ports when ERP ring is configured and will be restarted on ERP ring deletion on ERP Ports.
- STP will continue to be administratively disabled on ERP ring ports if ERP ring is disabled.

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 13-17](#) 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 13-16](#) for more information.

Source Learning

The ERP protocol determines and performs the MAC address flushing per port.

QoS Interface

The interaction between ERP and QoS is for the purpose so that R-APS PDUs can be handled appropriately by the switch.

MVRP

ERP NI must provide blocking or forwarding state of ERP ports to MVRP.

Quick Steps for Configuring ERv2 with Standard VLANs

The following steps provide a quick tutorial for configuring ERv2.

- 1** Create a ERP service VLAN with 8021Q VPAs on ring ports (on which ring needs to be created) prior to ring creation using the **vlan** command.

```
-> vlan 1001
-> vlan 1001 802.1q 1/3
-> vlan 1001 802.1q 1/4
```

The 802.1q binding has to be created prior to ring creation for SVLAN. For all other VLANs, as soon as 802.1q or default binding is created (except for service VLAN), the VPA is controlled by ERv2. As soon as 802.1q or default binding (except for service VLAN) is removed from ring port, ERv2 stops controlling the VPA.

- 2** Create ERP ring ID, 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** Enable the ERP ring configuration using the **erp-ring enable** command.

```
-> erp-ring 1 enable
```

- 5** Display the ERP configuration using the **show erp** command.

```
-> show erp
```

Quick Steps for Configuring ERv2 with VLAN Stacking

The following steps provide a quick tutorial for configuring ERv2 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 VLAN stacking Network Network Interface (NNI) ports, associate the ports with SVLAN, and configure them for use with ERP using the **ethernet-service svlan nni** command.

```
-> ethernet-service nni port 1/1
-> ethernet-service nni port 1/2
-> ethernet-service svlan 1001 nni port 1/1
-> ethernet-service svlan 1001 nni port 1/2
```

- 4 Create ERP ring ID 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 1002 nni port 1/1-2
-> ethernet-service svlan 1002 nni port 1/2-2
```

- 7 Enable the ERP ring configuration using the **erp-ring enable** command.

```
-> erp-ring 1 enable
```

- 8 Display the ERP configuration using the **show erp** command.

```
-> show erp
```

ERPV2 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 ERPv2 with Standard VLANs”](#) on page 13-10, [“Quick Steps for Configuring ERPv2 with VLAN Stacking”](#) on page 13-11

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 13-13.
- 2** Tag VLANs for ring protection. See [“Adding VLANs to Ring”](#) on page 13-14.
- 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 13-15.
- 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 13-15.
- 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 13-16.
- 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 13-16.
- 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 13-17.
- 8** Enable ERP virtual channel. See [“Enabling and Disabling R-APS Virtual Channel”](#) on page 13-18
- 9** Configure revertive mode. Different configurations related to revertive and non-revertive mode can be configured. See [“Configuring Revertive and Non-revertive Mode”](#) on page 13-19
- 10** Clear ERP statistics. Commands to clear ERP statistics for a single ring or multiple rings are described in [“Clearing ERP Statistics”](#) on page 13-21.

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.
- An ERP ring port can belong to only one ERP ring at a time.
- STP is automatically disabled when ERP is enabled on any port.
- If the ERP switch participates in an Ethernet OAM Maintenance Domain (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.

- Use the **erp-ring reset-version-fallback** command after upgrading OmniSwitch to the latest AOS version. This command must be issued on all nodes in a ring, starting from the RPL node as part of the ERPv2 upgradation process. Refer to the Release Notes for more information on the upgrade procedure.

Major and Sub-ring Management

A shared link must be configured only on the major ring.

The following conditions must be considered for configuring an ERPv2 port for a shared link:

- Sub-rings can not be closed using a shared link.
- An SVLAN must exist before an ERP ring is created and must be unique per ring.
- A given port can only be configured on one ring.
- Each ring must have its own RPL.
- The RPL can be placed anywhere on the major ring including the shared links.
- The RPL can be placed anywhere on the sub-rings, including the sub-ring-port. Since the sub-ring is not closed using the shared link, the RPL cannot be placed on the shared link.

Configuration Parameters

The following conditions must be considered before configuring an ERPv2 port:

- A given port can only be configured on one ring.
- The shared links are only configurable on the Master Ring.
- The sub-rings cannot be closed using the shared links.
- Each ring must have its own RPL.
- The RPL can be placed anywhere on the Master Ring, including the shared links.
- The RPL can be placed anywhere on the sub-rings, including the only ring port of the interconnection nodes. Since the sub-ring is not closed using the shared link, the RPL cannot be placed on the shared link.

Configuring an ERP Ring

The following configuration steps are required to create an ERP ring:

- 1** Determine which two ports on the switch become the ring ports. For example, ports 1/2 and 3/1.
- 2** Determine which VLAN on the switch becomes the ERP service VLAN for the ring. If the VLAN does not exist, create the VLAN. For example:

```
-> vlan 500
```

- 3** 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.

4 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
```

5 Repeat Steps 1 through 4 for each switch that participates 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 AOS Release 6 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.

Configuring an ERPV2 Sub-ring

An ERPV2 sub-ring can be configured on the interconnection node. Sub-ring port along with the service VLAN and MEG level will have to be specified. There will be only one port as a part of sub-ring on the interconnection node. Other ring ports will be a part of major ring and not the sub-ring.

A sub-ring on the interconnection node can be configured using the **erp-ring sub-ring-port** command:

```
-> erp-ring 3 sub-ring-port 1/3 service-vlan 10 level 2
```

A sub-ring on the non-interconnection node can be configured using the **erp-ring** command:

```
-> erp-ring 3 port1 1/1 port2 1/3 service-vlan 10 level 2
```

Adding VLANs to Ring

ERP allows a single VLAN or a number of VLANs to participate in a single ERP ring. The following configurations must be performed on each switch in the ERP ring network.

Step 1: Create the Service VLAN and add to ring ports.

```
-> vlan 10
-> vlan 200
-> vlan 10 802.1q port 1/3
-> vlan 10 802.1q port 1/5
-> vlan 200 802.1q port 1/6
```

Step 2: Create traffic VLANs and add to ring ports as necessary using the following commands:

```
-> vlan 100-400
-> vlan 100-300 802.1q 1/5
-> vlan 100-300 802.1q 1/3
-> vlan 201-400 802.1q 1/6
```

Note. The traffic VLANs can be added or deleted as needed at any time during the configuration. Once we configure a port as ERP ring port, STP is disabled on that port and ERPv2 is operational on all the VLANs tagged on that port.

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 serves 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 is 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 AOS Release 6 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 AOS Release 6 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 AOS Release 6 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 17, “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
Ring-Id : 1
  Ring Port Status      : forwarding,
  Ring Port Type        : non-rpl,
  Ethoam Event          : disabled
  Remote-endpoint Id    : none
```

For more information about these commands, see the *OmniSwitch AOS Release 6 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.

Note. Unless explicitly configured with a default VLAN other than VLAN1, the default VLAN on an NNI interface is 4095.

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** form 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
-> ethernet-service svlan 100 nni 1/2
-> 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 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.
- 4 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.

Enabling and Disabling R-APS Virtual Channel

Virtual channel can be enabled or disabled. By default, R-APS virtual channel is enabled.

Enabling R-APS Virtual Channel

Enable R-APS virtual channel using the following command:

```
-> erp-ring 2 virtual-channel enable
```

R-APS messages from the sub-ring on the interconnection node are forwarded as normal data to the major ring ports. A node is identified as interconnection node when at least one ring is configured with a sub-ring-port.

R-APS messages from the sub-ring are tagged with the sub-ring SVLAN, are forwarded to the major ring member ports of this SVLAN.

Note. All the ring ports in major ring must be member of the sub-ring SVLAN to support R-APS virtual channel.

Interconnection Node of the Sub-ring

When R-APS virtual channel is enabled, on the interconnection node of a sub-ring, all the R-APS messages received from sub-ring port are processed and flooded to major ring ports that are the member of the VLAN used by R-APS message.

For example,

```
-> erp-ring 3 virtual-channel enable
```

Other nodes of the Sub-ring

When enabled, R-APS messages received on blocked port are processed but not forwarded to the other ring port.

Disabling R-APS Virtual Channel

Disable R-APS virtual channel using the following command:

```
-> erp-ring 2 virtual-channel disable
```

Now, R-APS messages from the sub-ring on the interconnection node are not forwarded to any other ports. R-APS messages are forwarded even on the blocked ports in the sub-ring. A configuration object is required for the sub-ring to disable the R-APS virtual channel.

Interconnection Node of the Sub-ring

When virtual channel is disabled, R-APS message received from sub-ring ports are processed but not flooded to major ring.

For example,

```
-> erp-ring 3 virtual-channel disable
```

Other nodes of the Sub-ring

When virtual channel is disabled, R-APS messages received on blocked port are processed and forwarded to other ring port.

Note. Virtual channel configuration must be consistent among all nodes of the sub-ring.

Configuring Revertive and Non-revertive Mode

The following section provides details on the different configurations related to revertive and non-revertive mode configurations.

Enabling Revertive Mode

Revertive mode can be enabled using the following command. By default, revertive mode is enabled.

```
-> erp-ring 2 revertive enable
```

Non-revertive Mode

Under non-revertive mode, when the failure recovers, the blocked port stays blocked and the unblocked RPL stays unblocked.

When the ERPv2 node is operating with ERPv1 node in the same ring, it operates in different way for compatibility. In this mode, revertive mode is always assumed, it operates in revertive mode regardless of user configuration.

Disable the revertive mode by using the following command:

```
-> erp-ring 2 revertive disable
```

Clear Non-revertive and Revertive Mode

When the ring is in the No Request (NR) state and the blocked port is not the RPL port, the operator must be allowed to trigger the reversion to the initial state of the ring (make the RPL port blocked).

This situation happens in 2 cases:

- The ring is set in a non-revertive mode.
- The ring is set in a revertive mode but the WTR timer has not expired.

The CLI command is as follows:

```
-> erp-ring 2 clear
```

The command can only be issued on the RPL owner node and when the ring is in the NR state and WTR timer not expired or no WTR (non-revertive mode)

When the command is accepted, the RPL owner node blocks its RPL port, and transmits an R-APS (NR, RB) message in both directions. Upon receiving the R-APS (NR, RB), each node unblocks its blocking ports and performs a flush operation when applicable.

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 AOS Release 6 CLI Reference Guide*.

ERPV2 Application Example

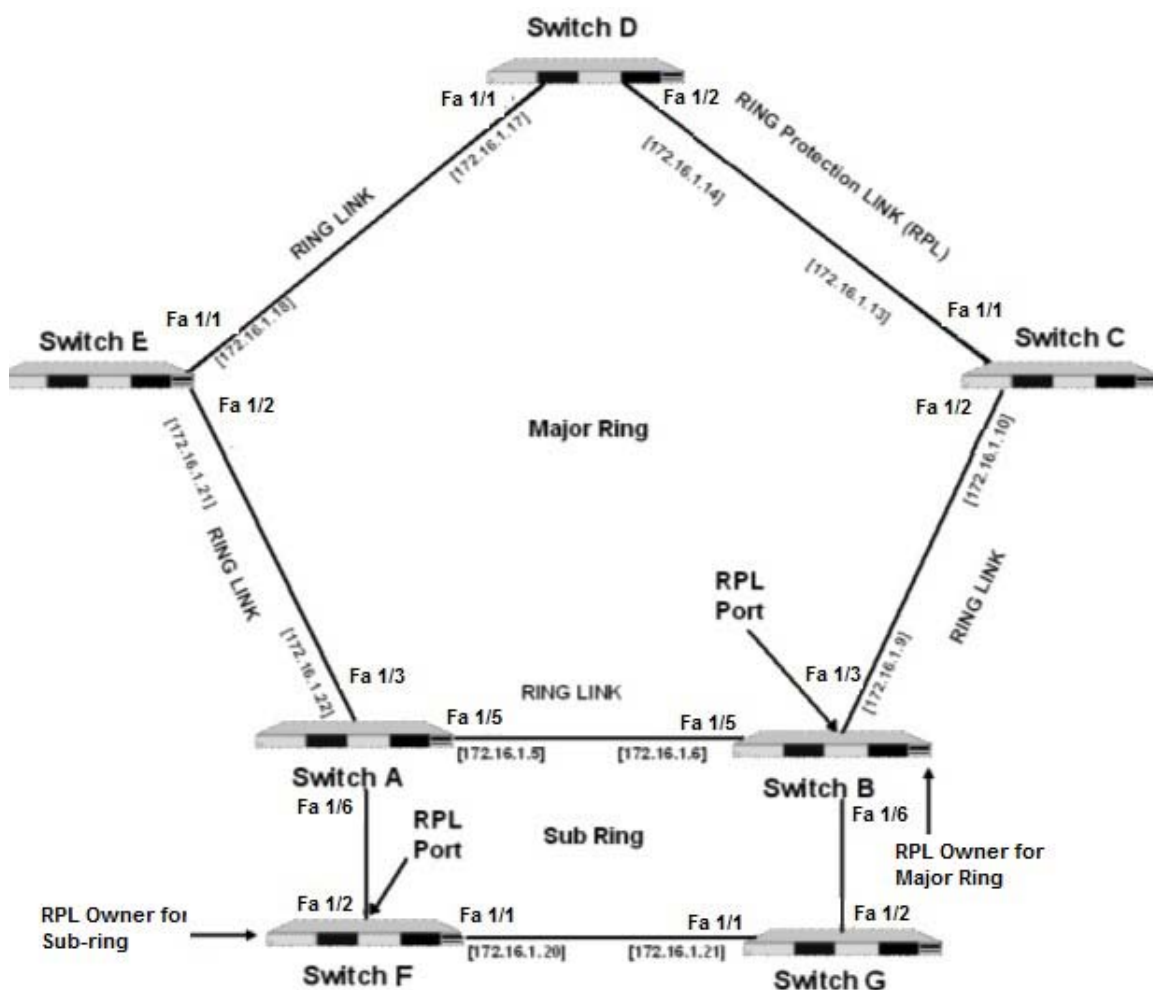
This section provides an example network configuration in which ERPv2 is configured on network switches to maintain a loop-free topology. In addition, a tutorial is also included that provides steps on how to configure the example network topology using the Command Line Interface (CLI).

ERPV2 Ring

The following diagram shows a seven-switch ERPv2 ring configuration when R-APS virtual channel is enabled.

The topology of the network is as follows:

- Switches A, B, C, D, and E for the major ring.
- Switch A and B form a shared link.
- Switch B is configured to be the main RPL node.
- Switches A, B, F, and G form the sub-ring.



The following sub-sections provide the details on prerequisites and different configurations for switches to set up an ERPV2 ring network.

Configuring the Shared Link

The following configurations must be performed on Switch A and Switch B.

Step 1: Create the Service VLAN and add to ring ports on Switch A and B that are part of a shared link:

```
Switch A -> vlan 10
Switch A -> vlan 200
Switch A -> vlan 10 802.1q 1/3
Switch A -> vlan 10 802.1q 1/5
Switch A -> vlan 200 802.1q 1/6
```

```
Switch B -> vlan 10
Switch B -> vlan 200
Switch B -> vlan 10 802.1q 1/3
Switch B -> vlan 10 802.1q 1/5
Switch B -> vlan 200 802.1q 1/6
```

Step 2: Create the ERP rings 1 and 2 on Switch A.

```
Switch A -> erp-ring 1 port1 1/5 port2 1/3 service-vlan 10 level 1
Switch A -> erp-ring 2 sub-ring-port 1/6 service-vlan 200 level 1
```

Step 3: Create traffic VLANs and add to ring ports as necessary using the following commands on Switch A.

```
Switch A -> vlan 100-400
Switch A -> vlan 100-300 802.1q 1/5
Switch A -> vlan 100-300 802.1q 1/3
Switch A -> vlan 201-400 802.1q 1/6
```

Step 4: Enable the rings on Switch A.

```
Switch A -> erp-ring 1 enable
Switch A -> erp-ring 2 enable
```

Configuring the Main RPL Node

Main RPL is configured on the Switch B. The following configurations must be performed on Switch B.

Step 1: Create the ERP rings 1 and 2 on Switch B.

```
Switch B -> erp-ring 1 port1 1/3 port2 1/5 service-vlan 10 level 1
Switch B -> erp-ring 2 sub-ring-port 1/6 service-vlan 200 level 1
```

Step 2: Configure Switch B as RPL Node using the **erp-ring rpl-node** command:

```
Switch B -> erp-ring 1 rpl-node 1/3
```

Step 3: Enable the rings on Switch B.

```
Switch B -> erp-ring 1 enable
Switch B -> erp-ring 2 enable
```

Step 4: Create traffic VLANs and add to ring ports as necessary using VLAN commands on Switch B.

```
Switch B -> vlan 100-400
Switch B -> vlan 100-300 802.1q 1/3
Switch B -> vlan 100-300 802.1q 1/5
Switch B -> vlan 201-400 802.1q 1/6
```

Configuring the Major Ring

The following configurations must be performed on Switch C, D, and E

```
-> vlan 10
-> vlan 10 802.1q 1/1
-> vlan 10 802.1q 1/2
-> erp-ring 1 port1 1/1 port2 1/2 service-vlan 10 level 1
-> vlan 100-300
-> erp-ring 1 enable
-> vlan 100-300 802.1q 1/1
-> vlan 100-300 802.1q 1/2
```

Configuring the Sub-ring and RPL Node

The following configurations must be performed on Switch F in the ERPV2 ring network:

```
-> vlan 200-400
-> vlan 200-400 802.1q 1/1
-> vlan 200-400 802.1q 1/2
-> erp-ring 2 port1 1/1 port2 1/2 service-vlan 200 level 1
-> erp-ring 2 rpl-node 1/2
-> erp-ring 2 enable
```

Configuring the Sub-ring

The following configurations must be performed on Switch G in the ERPV2 Ring network:

```
-> vlan 200-400
-> vlan 200-400 802.1q 1/1
-> vlan 200-400 802.1q 1/2
-> erp-ring 2 port1 1/2 port2 1/1 service-vlan 200 level 1
-> erp-ring 2 enable
```


Verifying the ERIPv2 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 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 AOS Release 6 CLI Reference Guide*.

14 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 (for example, 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 AOS Release 6 CLI Reference Guide*. This chapter provides an overview of LBD and includes the following information:

- [“LBD Specifications” on page 14-2](#)
- [“Quick Steps for Configuring LBD” on page 14-3](#)
- [“LBD Overview” on page 14-4](#)
- [“Configuring LBD” on page 14-6](#)
- [“LBD Use Case Scenario” on page 14-8](#)
- [“Verifying the LBD Configuration” on page 14-10](#)

LBD Specifications

| | |
|--------------------------|--|
| RFCs supported | NA |
| IEEE Standards Supported | NA |
| Platforms Supported | OmniSwitch 6350, 6450 |
| Ports Supported | There is no restriction on type of ports on which the LBD can be enabled. But it is recommended LBD has to 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 Transmission 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
```

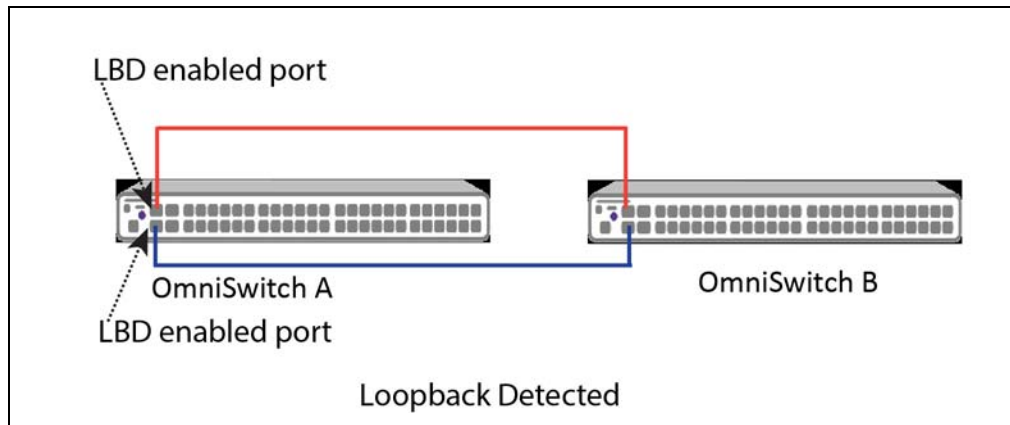
LBD Overview

Loopback Detection (LBD) automatically detects and prevents L2 forwarding loops on a port where STP cannot be used. Sometimes the STP based loop detection cannot be used due to the following facts:

- There is a client's equipment that drops or cuts the BPDUs.
- The STP protocol is restricted on edge Network.

The LBD feature detects that a port has been looped back or looped.

Note. When Loop Back Detection is enabled along with Spanning Tree Protocol (STP), the functionality of LBD is unpredictable. This is due to the fact that Loop Back on a switch depends on STP BPDU or LBD BPDU that is sent to trigger Loop avoidance.



If a LBD frame which is originated from a device is received back by the same device, the port in which the LBD frame is received will go to shutdown state and the port at the other end where it is connected to will go to inactive 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.

If Loopback detection is enabled globally or (globally and per port) on both the devices that are connected back to back as shown in above figure, loop will be detected.

The LBD frame generated from OmniSwitch A will be sent to OmniSwitch B. The OmniSwitch B will receive the LBD frame from OmniSwitch A and it will process the LBD frame, if LBD is enabled in the particular port. Otherwise the packet will be flooded. If both Source MAC (bridge MAC) and Destination MAC matches, the packet will be trapped to CPU. Otherwise the packet will be flooded.

When the LBD frame is received by the device from which it is originated, then loop will be detected and the port in which LBD is enabled will be shutdown.

The following are the scenarios which shows how loopback detection works depending on the global and port level configurations:

| OmniSwitch A | | OmniSwitch B | | Behaviour |
|----------------------|----------------------|----------------------|----------------------|---------------------------|
| LBD enabled globally | LBD enabled per port | LBD enabled globally | LBD enabled per port | |
| Yes | Yes | Yes | Yes | Loop will be detected |
| Yes | Yes | Yes | No | Loop will be detected |
| Yes | Yes | No | Yes | Loop will be detected |
| Yes | Yes | No | No | Loop will be detected |
| Yes | No | Yes | Yes | Loop will be detected |
| Yes | No | Yes | No | Loop will not be detected |
| Yes | No | No | Yes | Loop will not be detected |
| Yes | No | No | No | Loop will not be detected |
| No | Yes | Yes | Yes | Loop will be detected |
| No | Yes | Yes | No | Loop will not be detected |
| No | Yes | No | Yes | Loop will not be detected |
| No | Yes | No | No | Loop will not be detected |
| No | No | Yes | Yes | Loop will be detected |
| No | No | Yes | No | Loop will not be detected |
| No | No | No | Yes | Loop will not be detected |
| No | No | No | No | Loop will not be detected |

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 is no further transmission and receiving of any traffic on the blocked port. The port goes 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.

loopback-detection autorecovery-timer is a command used to recover the ports from shutdown mode. By default the value is 300 seconds same as ESM timer. Once the number of attempts specified in the ESM recovery timer in LBD is expired, the port moves to “permanent shutdown” state. After that, recovery timer will not recover the port.

To recover the interface from permanent shutdown state, use the command **interface clear-violation all**.

Permanent Shutdown for LBD Ports

When the ports are shut down due to loop and after the auto recovery timer expires, it recovers. Again if an LBD frame is encountered then the port goes to shutdown again. In the ESM, we have a maximum recovery time (300 secs) and recovery attempt (10) and LBD has auto recovery timer (300 secs default). Once the recovery attempts are reached, violation of port will move to "permanent shutdown" state. This state can be recovered only by clearing the interfaces violation manually.

To recover the interface from permanent shutdown state, use the command **`interface clear-violation all`**.

Note. When a port is shutdown due to LBD and loopback detection is disabled globally and on port level, the port will remain in down state until auto recovery timer expires. After that the port will be recovered.

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 is 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 14-7](#))
- Configure the LBD transmission timer (see [“Configuring the LBD Transmission Timer” on page 14-7](#))
- Configure the autorecovery timer (see [“Configuring the Autorecovery Timer” on page 14-7](#))
- View the LBD statistics on a port (see [“Viewing LBD Statistics” on page 14-7](#))
- Recover a port from LBD shutdown (see [“Recovering a Port from LBD Shutdown” on page 14-7](#))

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:

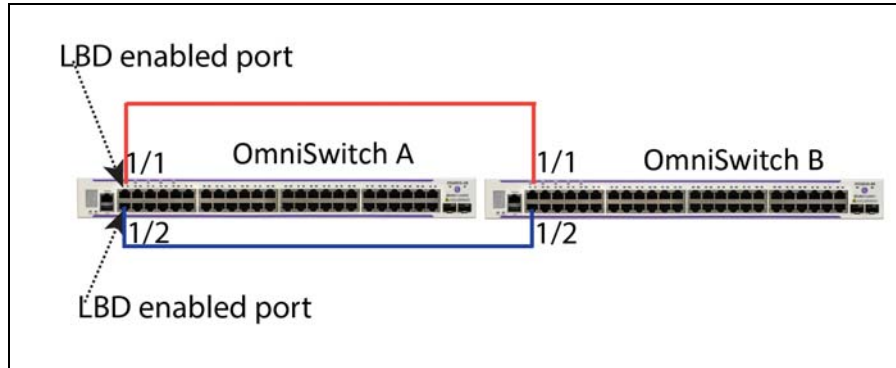
```
-> interfaces 1/5 clear-violation-all
```

To bring multiple ports out of the shutdown state, enter:

```
-> interfaces 5/5-10 clear-violation-all
```

LBD Use Case Scenario

In the following scenario, two switches (A and B) connected back to back with 2 ports.



Loopback Configuration

Configure two switches as follows:

1. Enable Loopback detection globally

```
-> loopback-detection enable
```

2. Verify Loopback detection globally

```
-> show loopback-detection
Global LBD Status           : enabled,
Global LBD Transmission Timer : 30 sec, level
Global LBD Auto-recovery Timer : 300 sec
```

3. Verify Loopback detection at port level

```
-> show loopback-detection port
Slot/Port   Admin State   OperState
-----+-----+-----
1/1         enabled      Normal
1/2         enabled      ShutDown
```

4. Verify Loopback detection statistics

```
-> show loopback-detection statistics port 1/1
LBD Port Statistics
LBD Packet Send           : 2
Invalid LBD Packet Received : 0
```

5. Clear Loopback detection statistics

```
-> clear loopback-detection statistics port 1/1
```

6. Verify Loopback detection statistics cleared

```
-> show loopback-detection statistics port 1/1
LBD Port Statistics
LBD Packet Send           : 0
Invalid LBD Packet Received : 0
```

7. Verify Loopback detection generated packets after clearing statistics

```
-> show loopback-detection statistics port 1/1
LBD Port Statistics
LBD Packet Send          : 3
Invalid LBD Packet Received : 0

-> show interfaces port
Legends: WTR - Wait To Restore
# - WTR Timer is Running & Port is in wait-to-restore state
* - Permanent Shutdown
```

| Slot/ Alias Port | Admin Status | Link Status | Violations | Recovery Time | Recovery Max | WTR (sec) |
|------------------------|-----------------|----------------|------------|------------------|-----------------|--------------|
| 1/11 | enable | up | none | 300 | 10 | 0 "" |
| 1/12 | enable | down | LBD | 300 | 10 | 0 "" |

8. Clear the violation caused due to Loopback detection

```
-> interfaces 1/1 clear-violation-all
```

9. Verify port release from Loopback detection violation

```
-> show loopback-detection port
Slot/Port  Admin State  OperState
-----+-----+-----
1/11      enabled     Normal
1/12      enabled     Normal
```

Note. When a port is shutdown due to LBD and loopback detection is disabled globally and on port level, the port will remain in down state until auto recovery timer expires. After that the port will be recovered.

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 AOS Release 6 CLI Reference Guide*.

15 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.

This implementation of CPE Test Head supports Unidirectional and Bidirectional, ingress tests. Traffic is generated at the UNI port as if the traffic was generated from a test head connected to the UNI port. This validates the actual customer SLA by subjecting the test traffic to the ingress QoS defined at the UNI port (Ethernet SAP profile or QoS policy rules for priority and bandwidth control) and the egress QoS defined at the egress NNI port and carrier network.

In unidirectional test, the test traffic is unidirectional. The traffic analysis is performed by the analyzer switch.

In bidirectional test, the test traffic is bidirectional. The traffic analysis is performed by the generator switch. The test traffic is sent to the generator switch using the hardware loopback function on the analyzer switch (Loopback switch).

The feature provides single-stream and multi-stream test capability.

The CPE Test Group (multi-stream) feature is supported on non-metro switches with metro license. The feature supports a stack containing up to 8 switches with a maximum of 8 test streams added to a test group.

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 ports, including CPE test ports, are subject to any SAP profile or QoS configuration associated with the port. This is important to consider when analyzing test results.

In This Chapter

This chapter describes the CPE Test Head feature, CPE Test Group 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 AOS Release 6 CLI Reference Guide*. This includes the following information:

- [“CPE Test Head Specifications” on page 15-2](#)
- [“Quick Steps for Configuring CPE Test Head” on page 15-3](#)
- [“CPE Test Head Overview” on page 15-5](#)
- [“CPE Test Head Configuration Overview” on page 15-6](#)
- [“Configuring a CPE Test Profile” on page 15-7](#)
- [“Configuring the L2 SAA Test” on page 15-9](#)
- [“Running a CPE Test” on page 15-10](#)
- [“Verifying the CPE Test Configuration and Results” on page 15-11](#)
- [“Configuring CPE Test Group” on page 15-13](#)
- [“CPE Test Head Advanced Configuration” on page 15-25](#)
- [“Sample Test Configurations” on page 15-27](#)

CPE Test Head Specifications

| | |
|---|--|
| Platforms Supported | OmniSwitch 6450 Metro license required for OmniSwitch 6450 |
| Tests supported | Unidirectional and bidirectional ingress test |
| Maximum number of test ID per switch | 32 |
| Number of active tests allowed per switch | 1 |
| Supported test roles | Generator or Analyzer or Loopback (Only one role per test; switch cannot perform more than one role for the same test). |
| Test mode supported | Ingress UNI |
| Test traffic direction supported | Unidirectional and bidirectional |

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 5 configure profile parameters common to both the generator and analyzer switch. Steps 6 through 8 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 test direction using the **test-oam direction** command. For example:

```
-> test-oam Test1 direction unidirectional
```

- 5 Configure the type of role the switch will perform using the **test-oam role** command. For example:

```
-> test-oam Test1 role generator
```

- 6 Configure the test port on the switch using the **test-oam port** command. For example:

```
-> test-oam Test1 port 1/1
```

- 7 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
```

- 8 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
```

For bidirectional test use the **fetch-remote-stats** parameter with the **test-oam start stop** command. For example:

```
-> test-oam Test1 start fetch-remote-stats
```

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 stop
```

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      Remote-Sys-Mac
-----+-----+-----+-----+-----+-----+-----
Test1  none  00:00:00:00:00:00  00:00:00:00:00:00  none  unidirectional  not-started  00:00:00:00:00:00
```

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  Remote-Stats  Throughput (Mbs)
-----+-----+-----+-----+-----+-----
Test1    19017       19017      19017       19017         9.98
```

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 AOS Release 6 CLI Reference Guide* for more information about these commands.

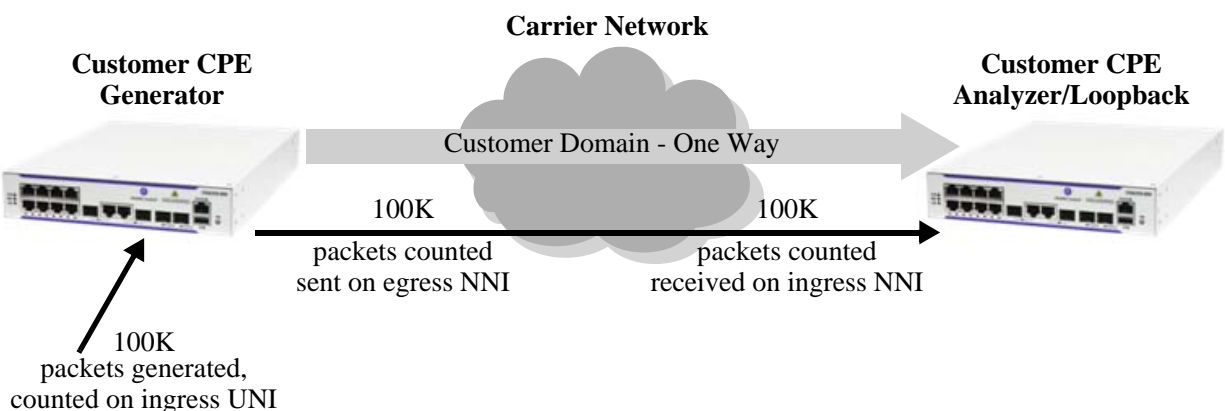
CPE Test Head Overview

The OmniSwitch CPE Test Head feature provides a remote test generator and analyzer/loopback 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/loopback.

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/loopback switch and then on the generator switch. The analyzer/loopback 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/loopback 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/ Loopback 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 or loopback) | 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 |
| Remote Sys MAC | Yes | No | test-oam remote-sys-mac |

Configuration Guidelines

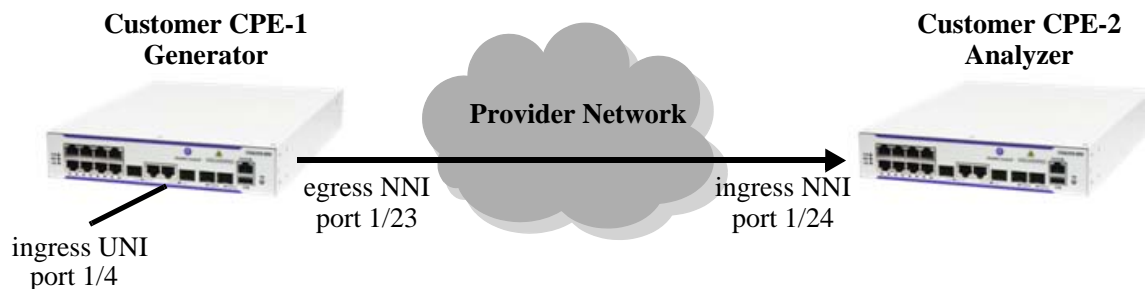
Consider the following guidelines when configuring the OmniSwitch CPE Test Head:

- Make sure the same test profile name (test ID) is used on the generator and analyzer/loopback switch.
- A switch can only perform one role (generator or analyzer or loopback) 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.
- For bidirectional test the role of the destination switch must be configured as loopback.
- Multicast and broadcast address must not be configured for bidirectional test.
- For the bidirectional test it is mandatory to configure the remote sys MAC address and activate the remote-fetch-stats option while starting the test.

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 bps 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 AOS Release 6 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 |

| CPE-1 Generator | CPE-2 Analyzer |
|--|---|
| <code>test-oam 100M_L2 role generator</code> | <code>test-oam 100M_L2 role analyzer</code> |
| <code>test-oam 100M_L2 port 1/4</code> | |
| <code>test-oam 100M_L2 duration 100 rate 100m packet-size 1518</code> | |
| <code>test-oam 100M_L2 frame vlan-tag 20 priority 5 ether-type 0x8101 data-pattern 0xabcd</code> | |

Refer to the *OmniSwitch AOS Release 6 CLI Reference Guide* for more information about these commands.

Configuring the L2 SAA Test

The L2-SAA test allows to measure the Round Trip Time (RTT) and Jitter during the test head operation. The L2-SAA test is performed between two OmniSwitch. The test can be run in parallel with the other CPE tests.

The L2-SAA test can also be configured for continuous monitoring of the network performance between the devices. The network performance is monitored by continuous injections of L2-SAA packets throughout the tests which generates and analyze network performance.

To configure the L2-SAA test, use the `test-oam l2-saa` command. For example:

```
-> test-oam test1 l2-saa priority 5 count 5 interval 1000 size 100 drop-eligible false
```

Note. The CPE test-oam string must be configured before using it in the L2-SAA test. The L2-SAA test derives the source MAC address, destination MAC address, and the VLAN ID from the test-oam configuration of the individual test frames.

To run the L2-SAA test continuously until the test-oam session ends, use the **continuous** parameter. For example:

```
-> test-oam test1 l2-saa continuous priority 5 interval 1000 size 100 drop-eligible false
```

On receiving the SAA reply for every frame, the minimum RTT, maximum RTT, total RTT, minimum Jitter, maximum Jitter, total Jitter and number of packets received will be calculated and stored in a global buffer to analyze the network performance between the devices.

Use the `show test-oam` command to view the L2-SAA configuration details.

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.
- **fetch-remote-stats**—Triggers the test at the remote device from the generator. The statistics are collected during the test and the test is stopped after receiving the test results. The **fetch-remote-stats** parameter must be used while starting a bidirectional test.

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
```

In case the test is a bidirectional test, the **fetch-remote-stats** parameter must be used. For example:

```
-> test-oam "test2" port 1/2 start fetch-remote-stats
```

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. |
| show test-oam saa statistics | Displays the SAA test statistics for all CPE tests or for a specific test name. |

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
Legend: Port: * = Inactive port

Total Test-Ids: 1
Test-Id Port Src-Mac          Dst-Mac          Vlan    Direction    Status    Remote-Sys-Mac
-----+-----+-----+-----+-----+-----+-----+-----
test1   1/5 00:11:22:12:44:55 00:22:33:12:44:55 1001    bidirectional  running

-> show test-oam Test2
Legend: dei-drop eligible indicator
TEST Parameters for Test2:
  Source Endpoint      : SW1,
  Destination Endpoint : SW2,
  Test Description     : IPV6 Test,
  Direction            : unidirectional,
  Source MAC           : 00:11:22:33:44:55,
  Destination MAC      : 00:22:33:44:55:66,
  Remote Sys MAC       : E8:E7:32:72:01:A4,
  Duration             : 10(secs),
  Vlan                 : 100,
  Role                 : generator,
  Port                 : 1/1,
  Tx Rate              : 8k,
  Frame Size           : 100,
  State                : start,
  Status               : running

  Frame Configuration :
    Frame Type         : ipv6,
    Vlan               : 200,
    Priority            : 7,
    Pattern            : 0x0001,
    Dei                : true,
    Source Ip          : 00:00:00:00:10.20.30.50,
    Destination Ip     : 00:00:00:00:10.30.40.60,
    Source Port        : 10,
    Destination Port   : 20,
    Next Header        : tcp,
    Hop-Count          : 50,
    Traffic-Class      : 0xff
    Flow-Label         : 0x0

  L2-SAA Configuration :
    L2-SAA Count       : 7
    L2-SAA Interval    : 1000
    L2-SAA DE          : TRUE
```

```
L2-SAA Payload Size      : 66
L2-SAA Priority          : 0
```

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  Remote-Stats  Throughput (Mbps)
-----+-----+-----+-----+-----+-----
Test1        1200366     1200366     0           1200366       8
Test2         0           0           1200366     1200366       8
Test3        95553      95553      95553       95553         7.33
```

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 for shows statistics for **RX-Ingress** because the switch is the analyzer for “Test2”. The “Test3” displays the statistics for **remote** test, the number of test frames received by the analyzer/loopback and fetched by the generator device. TX-Ingress, TX-Egress, RX-Ingress, Remote-Stats, and Throughput (Mbps). **Throughput (Mbps)**, displays the traffic throughput of the test.

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 AOS Release 6 CLI Reference Guide*.

Configuring CPE Test Group

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. The feature provides a multi-stream test capability. The CPE multi-test feature is supported on non-metro switches with metro license. The feature supports a stack containing up to eight switches.

The following information describes the CPE Test Group multi-test feature and how to configure it through the Command Line Interface (CLI):

- [“CPE Test Group Specifications” on page 15-13](#)
- [“Quick Steps for Configuring CPE Test Group” on page 15-14](#)
- [“CPE Test Group Overview” on page 15-17](#)
- [“CPE Test Group Configuration Overview” on page 15-18](#)
- [“Configuring a CPE Test Group Profile” on page 15-20](#)
- [“Running a CPE Test Group test” on page 15-22](#)
- [“Verifying the CPE Test Group Configuration and Results” on page 15-23](#)
- [“CPE Test Head Advanced Configuration” on page 15-25](#)
- [“Sample Test Configurations” on page 15-27](#)

CPE Test Group Specifications

| | |
|--|--|
| Platforms Supported | OmniSwitch 6350, 6450 Metro license required for OmniSwitch 6450. |
| Tests supported | Unidirectional and Bidirectional ingress test. |
| Maximum number of test-oam group per switch | 32 |
| Maximum number of test streams per test group | 8 |
| Number of active test-oam group allowed per-switch | 1 |
| Supported test roles | Generator or Analyzer or loopback (Only one role per test; switch cannot perform more than one role for the same test). |
| Test mode supported | Ingress UNI. |
| Test traffic direction supported | Unidirectional and Bidirectional. |

Quick Steps for Configuring CPE Test Group

The following steps provide a quick tutorial on how to configure a CPE test group and run the CPE test group. Each step describes a specific operation and provides the CLI command syntax that is used to perform that operation.

Configure the CPE Test Group Profile

The CPE test group profile is configured on both the generator and analyzer switch. Steps 2 through 6 configures profile parameters common to both the generator and analyzer switch. Steps 7 through 9 configures profile parameters required only for the generator.

- 1 Configure the feeder port globally in the system to feed the test traffic to generator port, use the **test-oam feeder** command. For example:

```
-> test-oam feeder-port 1/4
```

- 2 Configure the name for the CPE test group, use the **test-oam group** command. For example:

```
-> test-oam group Testgroup1 descr First-testgroup
```

- 3 Configure the list of tests that need to be added in the CPE test group, use the **test-oam group tests** command. For example:

```
-> test-oam group Testgroup1 tests test1 test2 test3 test4 test5 test6 test7 test8
```

- 4 Configure the source and destination end point for the CPE test group, use the **test-oam group src-endpoint dst-endpoint** command. For example:

```
-> test-oam group Testgroup1 src-endpoint SW1
```

```
-> test-oam group Testgroup1 dst-endpoint SW2
```

- 5 Configure the test direction using the **test-oam group direction** command. For example:

```
-> test-oam group Testgroup1 direction bidirectional
```

- 6 Configure the required role for the switch using the **test-oam group role** command. For example:

```
-> test-oam group Testgroup1 role generator
```

Note. For bidirectional test the role of the destination switch must be configured as loopback. Direction cannot be set Bidirectional when role is Analyzer and vice-versa.

- 7 Configure the CPE test group port on the generator switch using the **test-oam group port** command. For example:

```
-> test-oam group Testgroup1 port 1/2
```

- 8 Configure the CPE test group duration and rate using the **test-oam group duration rate** command. For example:

```
-> test-oam group Testgroup1 duration 10 rate 8 kbps
```

9 Configure the remote sys mac for the CPE group using the **test-oam group remote-sys-mac** command. This configuration is mandatory in case of bidirectional test. For example:

```
-> test-oam group "Testgroup1" remote-sys-mac E8:E7:32:32:A6:EE
```

Running the CPE Test Group test

1 Start the test on the analyzer switch first and then on the generator switch using the **test-oam group start** command. For example:

```
-> test-oam group Testgroup1 port 1/2 start
-> test-oam group Testgroup1 start
```

When the test runs for the amount of time specified in the test duration, the test automatically stops.

In case the test is a bidirectional test, the **fetch-remote-stats** parameter must be used. For example:

```
-> test-oam group Testgroup1 port 1/2 start fetch-remote-stats
```

2 To stop an active test from running, use the **test-oam group stop** command. For example:

```
-> test-oam group Testgroup1 stop
```

Note. Verify the CPE test group configuration and status with the **show test-oam group** command. For example:

```
-> show test-oam group tests
```

```
Total Test-Groups: 2
Feeder Port      : none
Test-Group  Port  Duration      Rate  Nb of  Direction  Status  Remote-Sys-Mac
                (secs)
-----+-----+-----+-----+-----+-----+-----+-----+
Testgroup1 none    5          -      2  unidirectional  not-started  00:00:00:00:00:00
Testgroup2 none    5          -      3  unidirectional  not-started  00:00:00:00:00:00
```

```
-> show test-oam group Testgroup1
```

Legend: Port: * = Inactive port

```
TEST Parameters for Testgroup1:
Source Endpoint      : SW1,
Destination Endpoint : SW2,
Test Group Description : first-testgroup,
Direction           : bidirectional,
Role                : generator,
Tx Rate             : 10m,
Duration            : 60 (secs),
Port                : 1/1,
State               : stop,
Status              : ended,
Remote Sys MAC      : E8:E7:32:32:A6:EE
Flow 1:
  Test Name          : test1,
  Vlan               : 1001,
  Tx Rate            : 10m,
  Source MAC         : 00:11:22:12:44:55,
```

```

Destination MAC      : 00:22:33:12:55:66,
Remote Sys MAC       : E8:E7:32:32:A6:EE,
Frame Size           : 100,
L2-SAA DE            : False,
L2-SAA Payload Size  : 100,
L2-SAA Count         : 5,
L2-SAA Interval     : 1000,
L2-SAA Priority      : 0
Flow 2:
Test Name            : test2,
Vlan                  : 1001,
Tx Rate              : 10m,
Source MAC           : 00:11:22:13:44:55,
Destination MAC      : 00:22:33:13:55:66,
Remote Sys MAC       : E8:E7:32:32:A6:EE,
Frame Size           : 100,
L2-SAA DE            : False,
L2-SAA Payload Size  : 100,
L2-SAA Count         : 5,
L2-SAA Interval     : 1000,
L2-SAA Priority      : 0

```

To verify test results, use the [show test-oam group statistics](#) command. For example:

```

-> show test-oam group statistics
Test-Group Flow   TX-Ingress  TX-Egress   RX-Ingress   Remote-Stats Throughput (Mbps)
-----+-----+-----+-----+-----+-----
Testgroup1 test1  309911      309911      309911      309911        4.13
Testgroup1 test2  309730      309730      309730      309730        4.13

-> show test-oam group saa statistics
Test-Group Flow   Time of last run      RTT  RTT  RTT  Jitter Jitter Jitter Packets Description
                  Min      Avg      Max  Min  Avg  Max   Sent  Rcvd
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
Testgroup1 test1  2001-01-10,05:02:59.0 108   111  114   2     3     4     5   first-test-
group

```

To clear test statistics, use the [clear test-oam group statistics](#) command. For example:

```

-> clear test-oam group Testgroup1 statistics
This clears all the statistics related to "Testgroup1".

-> clear test-oam group statistics
This will clear the statistics for all the groups configured.

```

See the *OmniSwitch AOS Release 6 CLI Reference Guide* for more information about these commands.

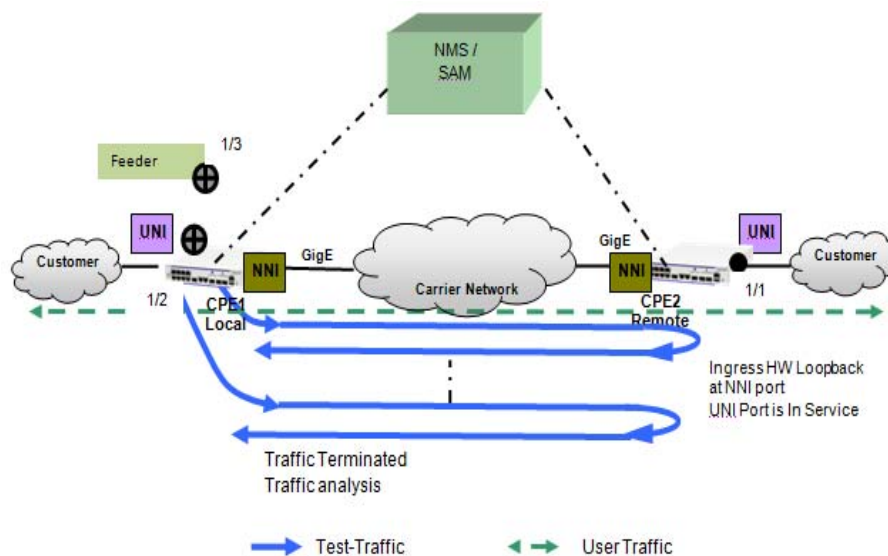
CPE Test Group Overview

The OmniSwitch CPE Test group feature provides a remote test generator and analyzer capability for testing and validating the Multi-CoS customer Ethernet service domain from end-to-end. The feature supports up to eight concurrent test flows. The OmniSwitch CPE Test group feature 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 group supports the ability to run unidirectional and bidirectional, ingress tests. Test setup involves configuring one CPE switch as the generator and a remote switch as the analyzer/loopback.

The following diagram shows an example of an OmniSwitch CPE Test Group configuration:



CPE Test group Example - Unidirectional, Ingress Test

In this example:

- 1 A feeder port must be configured in the system to feed the traffic to the generator. The feeder port is required while running a CPE test group.
- 2 The CPE test group 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.

3 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.

4 Traffic is counted and sent out on the SAP NNI port. This subjects the test frames to the egress NNI QoS policies.

5 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.

6 CPE Test group CLI **show** commands are used on the generator and analyzer switches to display and verify CPE test group 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 Group Configuration Overview

CPE Test Group 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 CPE test group 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.

| CPE Test group Parameters | Generator Switch | Analyzer/ Loopback Switch | CLI Command |
|---|------------------|---------------------------|---|
| Profile name | Yes | Yes | test-oam group |
| Source and destination endpoints | Yes | Yes | test-oam group src-endpoint dst-endpoint |
| Test-oam role (generator or analyzer or loopback) | Yes | Yes | test-oam group role |
| UNI port for test packet generation | Yes | No | test-oam group port |
| Test-oam duration and rate | Yes | No | test-oam group duration rate |
| Remote Sys MAC | Yes | No | test-oam group remote-sys-mac |

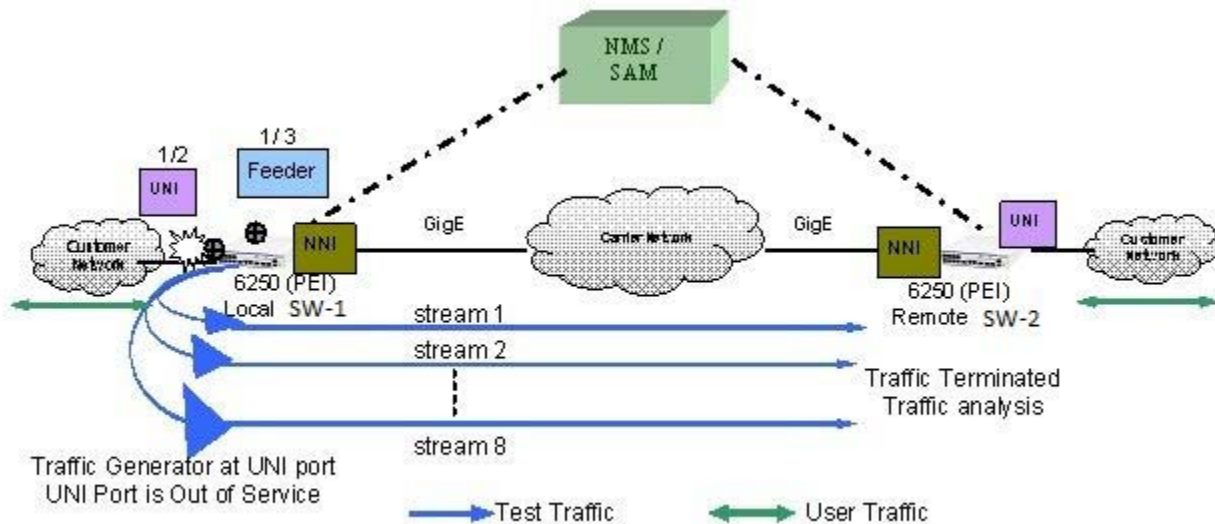
Configuration Guidelines

Consider the following guidelines when configuring the OmniSwitch CPE Test group:

- Make sure the same CPE test group 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.
- Each test which will be configured in the list of tests in the CPE test group that needs to run concurrently must be configured before adding in the list.
- Each flow is properly configured to be classified into the correct CoS or QoS profile.
- The sum of bandwidth of the grouped test streams must not exceed the supported line-rate of 100 Mbps for copper port and 1 Gig for fiber port.
- Only one CPE test group can be active for the switch at any given time.
- Up to 32 CPE test groups are allowed per switch.
- The feeder port must be configured to start a CPE test group.
- The VLAN used for a CPE test group must be a service VLAN.
- Each test in a CPE test group must have a unique VLAN, source mac-address, and destination mac-address.
- The modification to the test which is part of the active CPE test group is not allowed.
- The CPE test group supports eight-test flows that can run concurrently.
- For bidirectional test, the role of the destination switch must be configured as loopback.
- Multicast and broadcast address must not be configured for bidirectional test.

Configuring a CPE Test Group Profile

This section describes how to configure the following CPE test group example, which includes defining the CPE test group 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 group.



To configure the test setup shown in the above figure:

- 1 Configure the feeder port globally in the system to feed the test traffic to generator port, use the **test-oam feeder** command. For example:

```
-> test-oam feeder-port 1/3
```

The configured feeder port 1/3 will feed the test traffic from the CPE test group to the generator port.

- 2 Configure the CPE test group profile name and an optional description on the generator (SW-1 switch) and analyzer (SW-2 switch) using the **test-oam group** command. For example:

```
-> test-oam group Testgroup1 descr first-testgroup
```

When the “Testgroup1” CPE test group is created, a profile associated with this name is automatically created.

- 3 Configure the list of CPE test group tests that need to be added in the CPE test group using the **test-oam group tests** command. For example:

```
-> test-oam group Testgroup1 tests test1 test2 test3 test4 test5 test6 test7 test8
```

The configured list of CPE test group tests will run concurrently when the CPE test group Testgroup1 is started.

- 4** Configure the source (generator) and destination (analyzer) endpoints on SW-1 and SW-2 using the **test-oam group src-endpoint dst-endpoint** command. For example:

```
-> test-oam group Testgroup1 src-endpoint SW1 dst-endpoint SW2
```

The endpoint is identified using the DNS host name for the switch. In this example, “SW-1” and “SW-2” are the configured host names for the generator and analyze switch.

- 5** Configure SW-1 as the generator switch using the **test-oam group role** command. For example:

```
-> test-oam group Testgroup1 role generator
```

Use this command with the **generator** option on the SW-1 switch. This will configure the role parameter in the “Testgroup1” CPE test group profile that resides on SW-1.

- 6** Configure SW-2 as the analyzer switch using the **test-oam group role** command. For example:

```
-> test-oam group Testgroup1 role analyzer
```

Use this command with the **analyzer** option on the SW-2 switch. This will configure the role parameter in the “Testgroup1” CPE test group profile that resides on SW-2.

Note that a switch can only serve as the generator or the analyzer for any given test.

- 7** Configures the port in SW-1 on which the CPE test group test will run, using the **test-oam group port** command. For example:

```
-> test-oam group Testgroup1 port 1/2
```

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.

- 8** Configure the test duration and rate of the CPE test group packet on SW-1 using the **test-oam group duration rate** command. For example:

```
-> test-oam group Testgroup1 duration 20 rate 8m
```

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 group rate configuration is optional. The test bandwidth is considered by default if the group rate is not configured.

The following table provides a summary of the CLI commands used in the configuration example:

| SW-1 Generator | SW-2 Analyzer |
|--|--|
| test-oam group Testgroup1 descr first-testgroup | test-oam group Testgroup1 descr first-testgroup |
| test-oam group Testgroup1 tests test1 test2 test3 test4 test5 test6 test7 test8 | test-oam group Testgroup1 tests test1 test2 test3 test4 test5 test6 test7 test8 |
| test-oam group Testgroup1 src-endpoint SW1 dst-endpoint SW2 | test-oam group Testgroup1 src-endpoint SW1 dst-endpoint SW2 |
| test-oam group Testgroup1 role generator | test-oam group Testgroup1 role analyzer |
| test-oam group Testgroup1 duration 20 | test-oam group Testgroup1 duration 20 |
| test-oam group Testgroup1 port 1/2 | |
| test-oam group Testgroup1 rate 8m | |

Refer to the *OmniSwitch AOS Release 6 CLI Reference Guide* for more information about these commands.

Running a CPE Test Group test

A CPE test is started first on the analyzer switch and then on the generator switch using the **test-oam group start** command. For example:

```
-> test-oam group Testgroup1 start
```

This command also includes the following optional parameter used to specify runtime (active) values for the specified test:

port—the port on which the test will generate test frames.

When this runtime parameter is specified with the **test-oam group start** command, the parameter value is used instead of the value configured for the same parameter in the CPE test group profile. For example, if the “Testgroup1” profile specifies port 1/10 for the test, the following command will run the “Testgroup1” test on port 1/4:

```
-> test-oam group Testgroup1 port 1/4 start
```

Note. The runtime values specified for any of the optional **test-oam group 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 Group test

An active CPE test group 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 **test-oam group stop** command. For example:

```
-> test-oam group Testgroup1 stop
```

Stopping the CPE test group on both the generator and analyzer is recommended. The analyzer switch will 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 Group Configuration and Results

To display the CPE test group configuration and statistics information, use the **show** commands listed below:

- show test-oam group** Displays the configuration and status of the CPE test groups.
- show test-oam group statistics** Displays the statistics for all CPE test groups or for a specific CPE test group.
- show test-oam group** Displays the SAA test statistics for all CPE test groups or for a specific test name.

The **show test-oam group** command displays the configuration and status of the CPE test groups. For example:

```
-> show test-oam group tests
```

```
Total Test-Groups: 2
Feeder Port      : none
Test-Group Port  Duration      Rate    Nb of  Direction  Status      Remote-Sys-Mac
                (secs)
-----+-----+-----+-----+-----+-----+-----+-----
Testgroup1 none    5          -        2    unidirectional  not-started  00:00:00:00:00:00
Testgroup2 none    5          -        3    unidirectional  not-started  00:00:00:00:00:00
```

```
-> show test-oam group TestGroup2
```

```
TEST Parameters for TestGroup2:
  Source Endpoint: SW1,
  Destination Endpoint: SW2,
  Test Group Description: DEFAULT,
  Direction: unidirectional,
  Role: generator,
  Tx Rate : -,
  Duration : 20 (secs),
  Port: 1/2,
  State: stop,
  Status: stopped
```

```
Flow1:
  Test Name : test_1,
  Vlan: 1001
  Tx Rate   : 1M,
  Source MAC: 00:00:00:00:01:01,
  Destination MAC: 00:00:00:00:01:02,
  Remote Sys MAC : E8:E7:32:72:01:A4,
  Frame size: 64,
  L2-SAA DE           : False,
  L2-SAA Payload Size : 100,
  L2-SAA Count        : 5,
  L2-SAA Interval     : 1000,
  L2-SAA Priority      : 0
```

```
Flow2:
  Test Name : test_2,
  Vlan: 1002
```

```

Tx Rate      : 10M,
Source MAC: 00:00:00:00:02:01,
Destination MAC: 00:00:00:00:02:02,
Remote Sys MAC : E8:E7:32:72:01:A4,
Frame size: 1518,
L2-SAA DE           : False,
L2-SAA Payload Size : 100,
L2-SAA Count        : 5,
L2-SAA Interval     : 1000,
L2-SAA Priority      : 0

```

Flow3:

```

Test Name : test_3,
Vlan: 1003
Tx Rate: 15M,
Source MAC: 00:00:00:00:03:01,
Destination MAC: 00:00:00:00:03:02,
Remote Sys MAC : E8:E7:32:72:01:A4,
Frame size: 1518,
L2-SAA DE           : False,
L2-SAA Payload Size : 100,
L2-SAA Count        : 5,
L2-SAA Interval     : 1000,
L2-SAA Priority      : 0

```

Flow4:

```

Test Name : test_4,
Vlan: 1004
Tx Rate: 5M,
Source MAC: 00:00:00:00:04:01,
Destination MAC: 00:00:00:00:04:02,
Remote Sys MAC : E8:E7:32:72:01:A4,
Frame size: 1518,
L2-SAA DE           : False,
L2-SAA Payload Size : 100,
L2-SAA Count        : 5,
L2-SAA Interval     : 1000,
L2-SAA Priority      : 0

```

The **show test-oam group statistics** command displays the statistics for all CPE test groups or for a specific CPE test group. For example:

```
-> show test-oam group statistics
```

| Test-Group | Flow | TX-Ingress | TX-Egress | RX-Ingress | Remote-Stats | Throughput(Mbs) |
|------------|-------|------------|-----------|------------|--------------|-----------------|
| TestGroup1 | flow1 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow2 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow3 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow4 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow5 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow6 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow7 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup1 | flow8 | 19017 | 19017 | 0 | 19017 | 9.98 |
| TestGroup2 | flow1 | 19017 | 19017 | 0 | 0 | |
| TestGroup2 | flow2 | 19017 | 19017 | 0 | 0 | |
| TestGroup2 | flow3 | 19017 | 19017 | 0 | 0 | |
| TestGroup2 | flow4 | 19017 | 19017 | 0 | 0 | |
| TestGroup3 | flow1 | 19017 | 19017 | 0 | 0 | |
| TestGroup4 | flow8 | 19017 | 19017 | 0 | 19017 | 9.98 |

The packet counts displayed are based on the role the switch plays for the specific test. For example, “TestGroup1” 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 “TestGroup1”.

To verify the received test packet count for “TestGroup1”, use the **show test-oam group statistics** command on the analyzer switch.

Note. For more information about the resulting display from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

CPE Test Head Advanced Configuration

Running L2 SAA test

The CPE test can be used to measure the Round Trip Time (RTT) and Jitter by using the **test-oam l2-saa** command. The L2 SAA test will run along with the data traffic test. The test results are captured at the generator switch.

For example:

```
-> test-oam test1 l2-saa count 8 size 120 priority 6 interval 900 drop-eligible true
```

Note. Use the **show test-oam saa statistics** command to view the test results.

Configuring Remote Sys MAC

The CPE test allows configuring a remote device to receive the test OAM messages on the generator side. The generator device can gather the test OAM messages from the remote device and store it in the local data base.

Configuring the Remote Sys MAC is mandatory for bidirectional test and optional for unidirectional test.

In case of single stream test, use the **test-oam remote-sys-mac** command to configure the remote device to receive the test OAM messages. For example:

```
-> test-oam Test1 remote-sys-mac 00:e0:b1:7c:7a:fa
```

In case of multi stream test, use the **test-oam group remote-sys-mac** command to configure the remote device to receive the test OAM messages. For example:

```
-> test-oam group Testgroup1 remote-sys-mac 00:e0:b1:7c:7a:fa
```

Saving the test results on the /flash

The test results can be stored on the /flash directory of the switch. The test information is appended at the end of the default text file. Two files are used to maintain the test statistics on the /flash directory active file (testoamActiveStats.txt) and inactive file (testoamInactiveStats.txt).

The current test statistics will be stored in the active file. When there is no space in the active file to store the test statistics, the active file is made inactive and the inactive file is made active and the stats are written by overwriting the old data.

Use the **test-oam statistics flash-logging** command to enable storing of the test information on the /flash. For example:

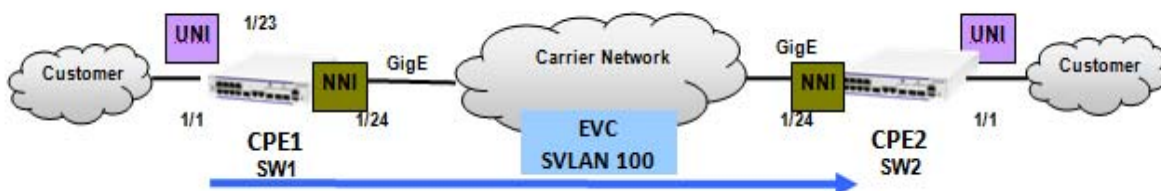
```
-> test-oam statistics flash-logging enable
```

Note. Use the **more** command to read the test results stored on the switch.

Sample Test Configurations

Sample Unidirectional Test Configuration

The following scenario represents a sample unidirectional test configuration:

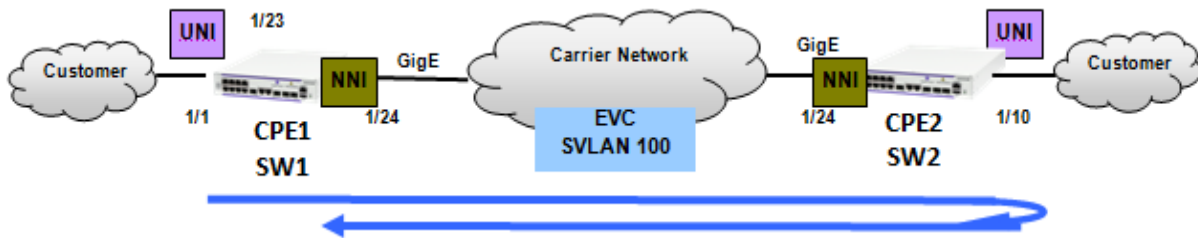


The CLI configuration for the displayed scenario is shown in the following table:

| CPE1 SW-1 Generator | CPE2 SW-2 Analyzer |
|--|--|
| <code>test-oam Test1</code> | <code>test-oam Test1</code> |
| <code>test-oam Test1 src-endpoint SW1</code> | <code>test-oam Test1 src-endpoint SW1</code> |
| <code>test-oam Test1 dst-endpoint SW2</code> | <code>test-oam Test1 dst-endpoint SW2</code> |
| <code>test-oam Test1 direction unidirectional</code> | <code>test-oam Test1 direction unidirectional</code> |
| <code>test-oam Test1 test-frame src-mac 00:00:00:00:00:01</code> | <code>test-oam Test1 test-frame src-mac 00:00:00:00:00:01</code> |
| <code>test-oam Test1 test-frame dst-mac 00:00:00:00:00:02</code> | <code>test-oam Test1 test-frame dst-mac 00:00:00:00:00:02</code> |
| <code>test-oam Test1 vlan 100</code> | <code>test-oam Test1 vlan 100</code> |
| <code>test-oam Test1 role generator</code> | <code>test-oam Test1 role analyzer</code> |
| <code>test-oam Test1 port 1/1</code> | |
| <code>test-oam Test1 frame vlan-tag 10 priority 5 ether-type ipv4 src-ip 1.1.1.1 dst-ip 2.2.2.2 src-port 2000 dst-port 4000</code> | <code>test-oam Test1 frame vlan-tag 10 priority 5 ether-type ipv4 src-ip 1.1.1.1 dst-ip 2.2.2.2 src-port 2000 dst-port 4000</code> |
| <code>test-oam Test1 packet-size 64</code> | |
| <code>test-oam Test1 rate 10Mbps</code> | |
| <code>test-oam Test1 duration 10</code> | |

Sample Bidirectional Test Configuration

The following scenario represents a sample bidirectional test configuration:



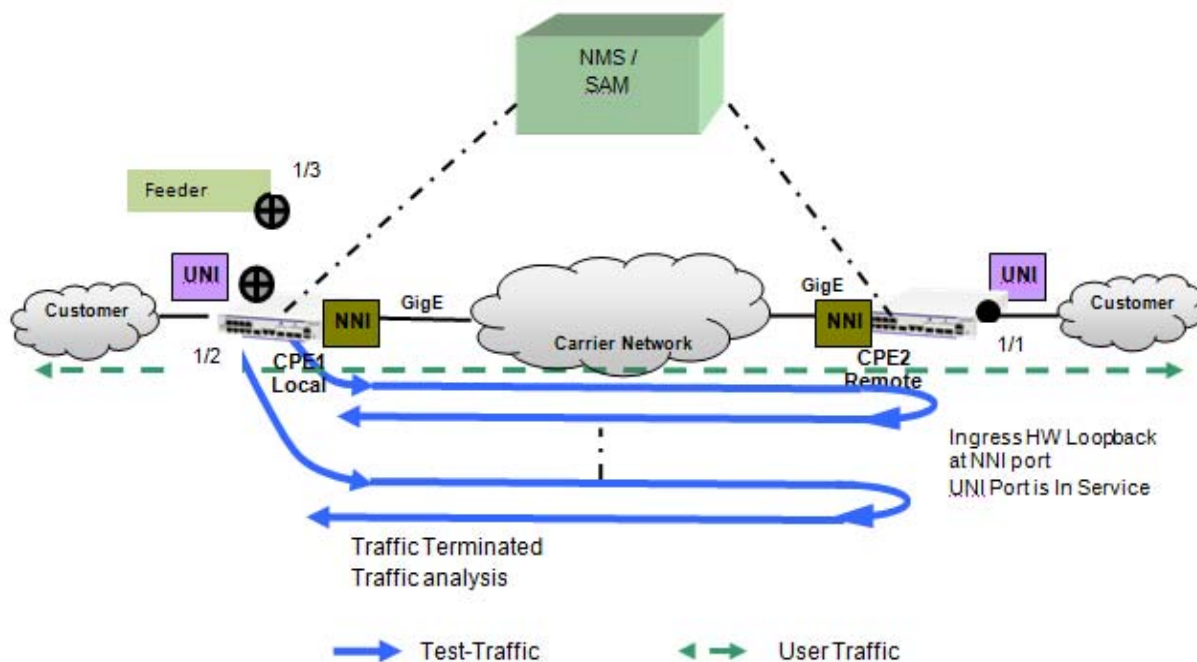
The CLI configuration for the displayed scenario is shown in the following table:

| CPE1 SW-1 Generator | CPE2 SW-2 Loopback |
|--|--|
| <code>test-oam Test2</code> | <code>test-oam Test2</code> |
| <code>test-oam Test2 src-endpoint SW1</code> | <code>test-oam Test2 src-endpoint SW1</code> |
| <code>test-oam Test2 dst-endpoint SW2</code> | <code>test-oam Test2 dst-endpoint SW2</code> |
| <code>test-oam Test2 direction bidirectional</code> | <code>test-oam Test2 direction bidirectional</code> |
| <code>test-oam Test2 test-frame src-mac 00:00:00:00:00:01</code> | <code>test-oam Test2 test-frame src-mac 00:00:00:00:00:01</code> |
| <code>test-oam Test2 test-frame dst-mac 00:00:00:00:00:02</code> | <code>test-oam Test2 test-frame dst-mac 00:00:00:00:00:02</code> |
| <code>test-oam Test2 vlan 100</code> | <code>test-oam Test2 vlan 100</code> |
| <code>test-oam Test2 role generator</code> | <code>test-oam Test2 role loopback</code> |
| <code>test-oam Test2 remote-sys-mac E8:E7:32:32:A6:EE</code> | <code>test-oam Test2 port 1/10</code> |
| <code>test-oam Test2 port 1/1</code> | <code>test-oam Test2 frame vlan-tag 10 priority 5 ether-type ipv4 src-ip 1.1.1.1 dst-ip 2.2.2.2 src-port 2000 dst-port 4000</code> |
| <code>test-oam Test2 frame vlan-tag 10 priority 5 ether-type ipv4 src-ip 1.1.1.1 dst-ip 2.2.2.2 src-port 2000 dst-port 4000</code> | <code>test-oam Test2 port 1/10</code> |
| <code>test-oam Test2 packet-size 64</code> | |
| <code>test-oam Test2 rate 10Mbps</code> | |
| <code>test-oam Test2 duration 10</code> | |

Note. For bidirectional test, the role of the destination switch must be configured as loopback.

Sample Bidirectional Multi-stream Test Configuration

The following scenario represents a sample bidirectional multi-stream test configuration:



The CLI configuration for the displayed scenario is shown in the following table:

| CPE1 SW-1 Generator | CPE2 SW-2 Loopback |
|---|---|
| test-oam feeder-port 1/3 | |
| test-oam group "testgroup1" descr "first-testgroup" | test-oam group "testgroup1" descr "first-testgroup" |
| test-oam group "testgroup1" tests "test2" "test3" | test-oam group "testgroup1" tests "test2" "test3" |
| test-oam group "testgroup1" direction bidirectional | test-oam group "testgroup1" direction bidirectional |
| test-oam group "testgroup1" src-endpoint SW1 dst-endpoint SW2 | test-oam group "testgroup1" src-endpoint SW1 dst-endpoint SW2 |
| test-oam group "testgroup1" remote-sys-mac E8:E7:32:32:A6:EE | test-oam group "testgroup1" remote-sys-mac e8:e7:32:32:a8:9e |
| test-oam group "testgroup1" port 1/2 | test-oam group "testgroup1" port 1/1 |
| test-oam group "testgroup1" role generator | test-oam group "testgroup1" role loopback |
| test-oam group "testgroup1" duration 10 | test-oam group "testgroup1" duration 10 |
| test-oam group "testgroup1" tests "test1" "test2" "test3" "test4" | test-oam group "testgroup1" tests "test1" "test2" "test3" "test4" |

Note. The individual tests must be configured before being added to the test group.
A maximum of eight test can be added in a group.

16 Configuring PPPoE Intermediate Agent

Point-to-Point Protocol over Ethernet (PPPoE) provides the ability to connect a network of hosts over a simple bridging access device to a Remote Access Concentrator (RAC). For example, Broadband Network Gateway. In PPPoE model, each host utilizes its own Point-to-Point Protocol (PPP) stack and the user is presented with a familiar user interface. Access control, billing, and type of service can be configured on a per-user, rather than a per-site, basis.

PPPoE Intermediate Agent (PPPoE-IA) solution is designed for the PPPoE access method and is based on the access node implementing a PPPoE-IA function to insert the access loop identification.

In This Chapter

This chapter describes the PPPoE-IA 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 AOS Release 6 CLI Reference Guide*.

This chapter includes the following:

- [“PPPoE-IA Specifications” on page 16-2](#)
- [“PPPoE-IA Defaults” on page 16-2](#)
- [“Quick Steps for Configuring PPPoE-IA” on page 16-3](#)
- [“PPPoE Intermediate Agent Overview” on page 16-5](#)
- [“Configuring PPPoE-IA” on page 16-6](#)
- [“Verifying PPPoE-IA Configuration” on page 16-9](#)

PPPoE-IA Specifications

| | |
|--|--|
| Platforms Supported | OmniSwitch 6350, 6450 Metro license required for OmniSwitch 6450 Enterprise models. |
| Maximum number of options supported for Circuit-Identifier | 5 |
| Maximum Circuit-Identifier length supported | 63 Bytes |
| Maximum Remote-Identifier length supported | 63 Bytes |

PPPoE-IA Defaults

Following are the PPPoE-IA default values:

| Parameter Description | Command | Default Value |
|--------------------------------|--|--------------------------------------|
| PPPoE-IA globally and on ports | <code>pppoe-ia</code> <code>pppoe-ia port {enable disable}</code> | Disabled |
| PPPoE-IA port | <code>pppoe-ia port {trust client}</code> | Client |
| Access-Node-Identifier | <code>pppoe-ia access-node-id</code> | Base MAC address of the switch |
| Circuit-ID | <code>pppoe-ia circuit-id</code> | “:” (colon) is used as the delimiter |
| Remote-ID | <code>pppoe-ia remote-id</code> | Base MAC address of the switch |

Quick Steps for Configuring PPPoE-IA

The following steps provide a quick tutorial on how to configure PPPoE-IA. Each step describes a specific operation and provides the CLI command syntax for performing that operation.

- 1 Enable PPPoE-IA globally on the switch using the **pppoe-ia** command.

```
-> pppoe-ia enable
```

Note. All PPPoE-IA parameters are configurable irrespective of the global status of PPPoE-IA. It is mandatory to enable PPPoE-IA globally as well as on a port for the PPPoE-IA feature to function.

- 2 Enable PPPoE-IA on a port or a link aggregate port using the **pppoe-ia port** command. For example, the following command enables PPPoE-IA on port 1/1 of the switch.

```
-> pppoe-ia port 1/1 enable
```

- 3 Configure a port or a link aggregate port as trusted or client port for PPPoE-IA using the **pppoe-ia port {trust | client}** command. By default, all ports are client ports. For example, the following command configures port 1/1 as a trusted port.

```
-> pppoe-ia port 1/1 trust
```

Note. The port that is connected to the PPPoE server must be configured as trusted, whereas the port connected to the host must be configured as a client port. Both client and trust ports must be in the same VLAN.

- 4 Configure a format to form an identifier that uniquely identifies an access node globally using the **pppoe-ia access-node-id** command. For example, the following command uses the base MAC address of the switch to identify an access node.

```
-> pppoe-ia access-node-id base-mac
```

- 5 Configure a Circuit-ID format that forms an identifier that uniquely identifies an access node globally, and an access loop that receives the PADI/PADR/PADT from the user side using the **pppoe-ia circuit-id** command. For example, the following command uses the base MAC address in ASCII format as the Circuit-ID.

```
-> pppoe-ia circuit-id ascii base-mac vlan
```

6 Configure a format to form an identifier that uniquely identifies the user attached to the access loop globally using the **pppoe-ia remote-id** command. For example, the following command uses the user configured string as the format for Remote-ID:

```
-> pppoe-ia remote-id user-string "remote-id-1"
```

Note. To view the global configuration for PPPoE-IA, enter the **show pppoe-ia configuration** command. The PPPoE-IA configuration is displayed as shown:

```
-> show pppoe-ia configuration
Status                               : enabled,
Access Node Identifier
  Access-node-id Format               : system-name,
  Access-node-id String              : vxTarget,
Circuit Identifier
  Circuit-Id Format                   : ascii,
  Circuit-id Field1                  : system-name,
  Circuit-id Field1 String           : vxTarget,
  Circuit-id Field2                  : base-mac,
  Circuit-id Field2 String           : 00:d0:95:ee:fb:02,
  Circuit-id Field3                  : interface,
  Circuit-id Field3 String           : ,
  Circuit-id Field4                  : none,
  Circuit-id Field4 String           : ,
  Circuit-id Field5                  : none,
  Circuit-id Field5 String           : ,
  Circuit-id Delimiter               : "|",
Remote Identifier
  Remote-id Format                    : mgnt-address,
  Remote-id String                   : 172.21.161.106
```

PPPoE Intermediate Agent Overview

PPPoE Intermediate Agent (PPPoE-IA) solution is designed for the PPPoE access method and is based on the access node implementing a PPPoE intermediate agent function to insert the access loop identification.

Access Node: An access node provides connectivity between the user and the network cloud. Access node aggregates the traffic coming from a user and routes it to the network. In the context of PPPoE-IA, an access node is the switch where the Intermediate Agent (IA) resides.

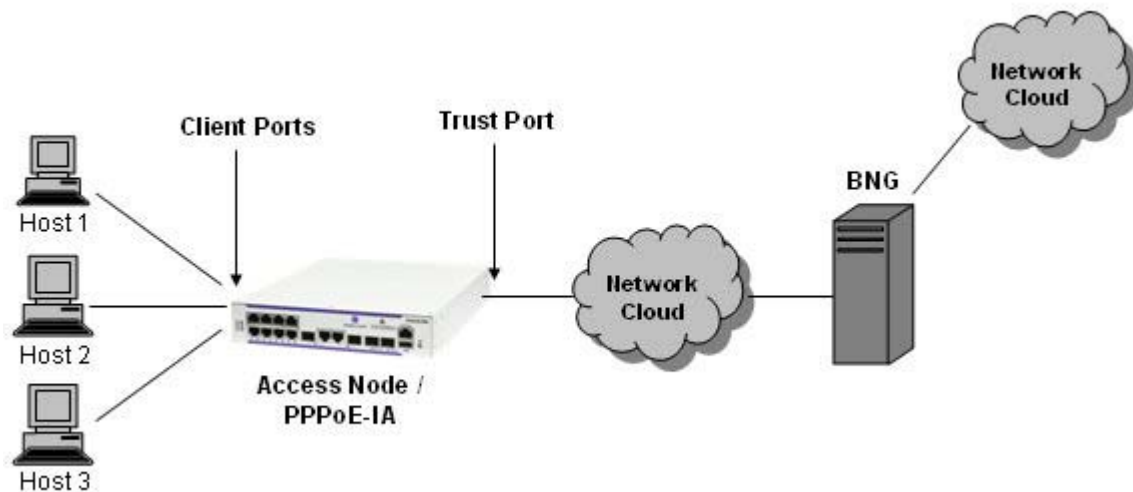
Access Loop: Access loop signifies the physical connectivity between the Network Interface Device (NID) at the customer premises and the access node. If a user is directly connected to the access node, the access loop can be identified by the interface number (slot/port). If the user is not directly connected or multiple users are connected to the access node through a single port, access loop for a particular user can be identified as the combination of interface (slot/port) and customer VLAN (CVLAN).

How PPPoE-IA Works

PPPoE-IA is a means by which the discovery packets of PPPoE are tagged at the access switch of the service provider using Vendor Specific Attributes (VSA) to add the line-specific information at the switch.

The purpose of an IA is to help service provider and the Broadband Network Gateway to distinguish between different end hosts connected over Ethernet to the access switch. The Ethernet frames from different users are appropriately tagged by the IA to provide this distinction. The AOS implementation of PPPoE-IA enables the rate limiting and insertion of VSA tags into the PPPoE Active Discovery (PAD) messages. The tag is allowed to contain information such as the base MAC address of the switch, interface, customer VLAN, system name, and a user-defined string depending on the configuration.

The following example illustrates the network overview for PPPoE IA.



Network overview for PPPoE IA

Configuring PPPoE-IA

This section describes how to configure PPPoE-IA using the CLI commands.

Enabling PPPoE-IA Globally

Enable the PPPoE-IA globally on the switch. By default, PPPoE-IA is disabled globally on the switch.

To enable PPPoE-IA globally on the switch, enter the `pppoe-ia` command at the CLI prompt as shown:

```
-> pppoe-ia enable
```

To disable PPPoE-IA globally on the switch, use disable option as shown:

```
-> pppoe-ia disable
```

Note. All PPPoE-IA parameters are configurable irrespective of the global status of PPPoE-IA. It is mandatory to enable PPPoE-IA globally as well as on a port for the PPPoE-IA to function.

Enabling PPPoE-IA on a Port

Enable or disable PPPoE-IA on a port or a link aggregate port by using `pppoe-ia {port | linkagg}` command. It is mandatory that PPPoE-IA is enabled globally as well as on a port.

For example, to enable PPPoE-IA on port 1/1 of the switch, enter:

```
-> pppoe-ia port 1/1 enable
```

To disable PPPoE-IA on a port 2/4, enter:

```
-> pppoe-ia port 2/4 disable
```

Note. PPPoE-IA is not supported on port mirroring destination ports, however, the configurations are accepted. PPPoE-IA is not supported on aggregable ports.

Configuring a Port as Trust or Client

Use `pppoe-ia {trust | client}` command to configure a port or a link aggregate port as trusted or client port. PPPoE-IA must be enabled on a client port as well as a trusted port for the feature to function. By default, all ports are client ports.

The port that is connected to the PPPoE Server must be configured as trusted, whereas the port connected to the host must be configured as a client port.

For example, to configure port 1/1 as a trusted port, enter:

```
-> pppoe-ia port 1/1 trust
```

For example, to configure link aggregate port 0 as a client port, enter:

```
-> pppoe-ia linkagg 0 client
```


Configuring Access Node Identifier for PPPoE-IA

To configure a format to form an identifier that uniquely identifies an access node, use the [pppoe-ia access-node-id](#) command.

For example, the following command uses the base MAC address of the switch to identify an access node:

```
-> pppoe-ia access-node-id base-mac
```

For example, the following command uses the user configured string to identify an access node:

```
-> pppoe-ia access-node-id user-string accessnode1
```

If the management address format is used as the Access Node Identifier, then the IP address of the Loopback0 interface (if configured and active) or the first active IP interface address is used as the management address. If none of them are available, IP address '0.0.0.0' is used as management address.

The access-node-identifier can have a maximum of 32 characters. The access-node-identifier longer than 32 characters is truncated to 32 characters when encoded in the VSA tag.

Configuring Circuit Identifier

The [pppoe-ia circuit-id](#) command globally configures a Circuit-ID format that forms an identifier that uniquely identifies an access node and an access loop on which the PPPoE Active Discovery Initiation (PADI) or PPPoE Active Discovery Request (PADR) or PPPoE Active Discovery Terminate (PADT) is received.

For Circuit-ID, two-format types are supported: default and ascii. The Circuit-ID is formed depending on the format as follows:

Default Circuit ID

default: When the PPPoE Circuit-ID is configured as default, the access-node-id is formed from either of the four supported formats: base-mac, system-name, mgnt-address, or user configurable string.

For example, the following command is used to configure the Circuit-ID as default.

```
-> pppoe-ia circuit-id default
```

When the Circuit-ID is configured as default, the Circuit-ID format in the Circuit-Identifier will display as "ethernet". For more information, see [show pppoe-ia configuration](#) command in the *OmniSwitch AOS Release 6 CLI Reference Guide*

default ATM: When the PPPoE-IA Circuit-ID format is configured as "default atm" the Circuit-ID encoding happens for "ATM" (Asynchronous Transfer Mode) parameter along with ethernet parameter.

For example, the following command is used to configure the Circuit-ID as "default ATM".

```
-> pppoe-ia circuit-id default atm
```

When the Circuit-ID is configured as default ATM, the Circuit-ID format in the Circuit-Identifier will display as "atm". For more information, see [show pppoe-ia configuration](#) command in the *OmniSwitch AOS Release 6 CLI Reference Guide*

ASCII Circuit ID

In the ascii Circuit-ID, the fields (maximum of five) are separated by delimiter up to a maximum of 63 characters.

For example, the following command uses the base-mac in ASCII format of the Circuit-ID:

```
-> pppoe-ia circuit-id ascii base-mac vlan
```

Configuring Remote Identifier

The Remote-ID identifies the host attached to the access loop. In AOS implementation, the Remote-ID identifies the access-node (that is, the IA).

The **pppoe-ia remote-id** command globally configures a format to form an identifier that uniquely identifies the user attached to the access loop.

For example, to use the base MAC address as the format for Remote-ID, enter:

```
-> pppoe-ia remote-id base-mac
```

If the management address format is used as the Remote-ID, the IP address of the Loopback0 interface (if configured and active) or the first active IP interface address is used as the management address. If none of them are available, IP address '0.0.0.0' is used as management address.

Verifying PPPoE-IA Configuration

A summary of the commands used for verifying the PPPoE-IA configuration is given here:

| | |
|------------------------------------|---|
| show pppoe-ia configuration | Displays the global configuration for PPPoE-IA. |
| show pppoe-ia | Displays the PPPoE-IA configuration for a port, port range or all the ports. |
| show pppoe-ia statistics | Displays the PPPoE-IA statistics for a port, link aggregate port, port range, or all the ports. |

To clear the statistics for all the physical or link-aggregate ports, a single port or a link aggregate port, or a range of physical ports for PPPoE-IA, use the **clear pppoe-ia statistics** command.

For more information about the output details that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

17 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 AOS Release 6 CLI Reference Guide*.

The following procedures are described in this chapter:

- [“Ethernet OAM Overview” on page 17-3.](#)
- [“Elements of Service OAM” on page 17-3.](#)
- [“Fault Management” on page 17-6.](#)
- [“Performance Monitoring” on page 17-6.](#)
- [“Interoperability with ITU-T Y.1731” on page 17-8.](#)
- [“Configuring Ethernet OAM” on page 17-10.](#)
- [“Verifying the Ethernet OAM Configuration” on page 17-17.](#)
- [“CVLANs for EVC MEPs” on page 17-14.](#)

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> [OmniSwitch supports only one-way delay (1DM) and two-way delay (DDM)] |
| Platforms Supported | OmniSwitch 6450 Metro license required for OmniSwitch 6450 Enterprise models. |
| Maximum Maintenance Domains (MD) per Bridge | 8 |
| Maximum Maintenance Associations (MA) per Bridge | 64 |
| Maximum Maintenance End Points (MEP) per Bridge | 128 |
| Maximum Remote Maintenance End Points (MEP) | 256 |
| 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 |

| Parameter Description | Command | Default Value/Comments |
|--|---|------------------------|
| 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 |
| Timeout value for Fault Notification Alarm Generation | ethoam fault-alarm-time | 2.5 seconds |
| Timeout value for Fault Notification Generation Reset | ethoam fault-reset-time | 10 seconds |

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 can isolate a fault down to a specific service, but to determine exactly where the fault occurred within the network infrastructure also requires 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 two or more MEPs.
- Point-to-point MA: logical sub-MA component only between two MEPs MA.
- Maintenance Domain (MD): One or more MAs under the same administrative control.

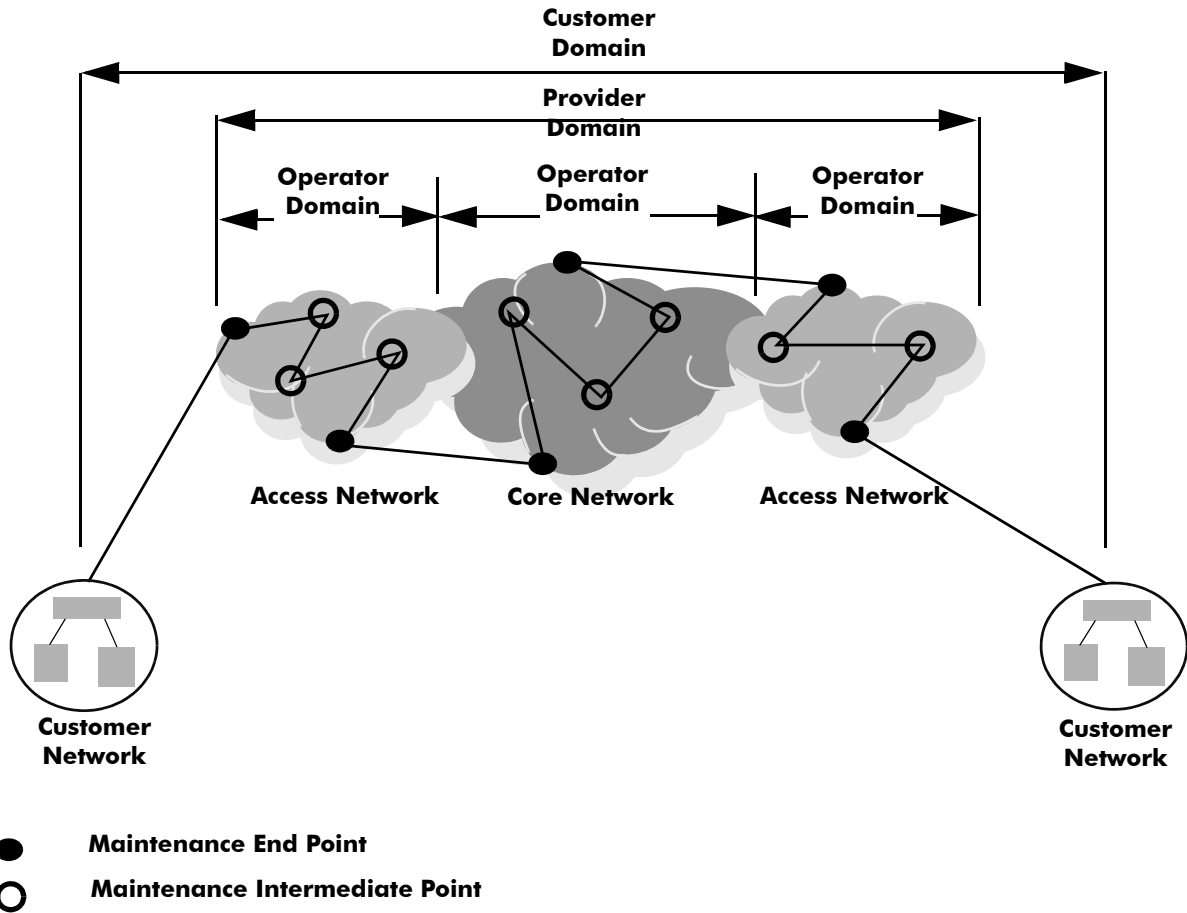
- Maintenance Domain Levels: There are 8 levels defined in 802.1ag:
 - levels [0, 1, 2] are for operators,
 - levels [3, 4] are for service provider
 - levels [5, 6, 7] are for customersMultiple levels are supported for flexibility.
- Mechanisms: continuity check (CC), loopback, link trace

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 operators (levels 0, 1, 2), service providers (levels 3, 4), and customers (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.

Any MEP can initiate or reply to an ETH-DM request, depending on the type of delay measurement requested. However during the phase of exchanging messages, five defects can be encountered by a MEP.

- Cross-connect defect
- Error CCM defect
- Remote defect
- MAC defect
- RDI defect

These defects are logged whenever there is a loss in connectivity between two connected MEPs. This feature is supported only on Metro models.

MIP CCM Database Support

Per section 19.4 of the IEEE 802.1ag standard, an MHF can 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 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.

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 AOS Release 6 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 17-10](#) 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.com format dnsName level 1
```

- 2 Create an Ethernet OAM Maintenance Association using the **ethoam association** command. For example:

```
-> ethoam association alcatel-lucent-sales format string domain esd.alcatel.com primary-vlan 10
```

- 3 Create an Configure the endpoint list for the Ethernet OAM Maintenance End Point Association using the **ethoam endpoint admin-state ethoam association endpoint-list** command. For example:

```
-> ethoam association alcatel-lucent-sales domain esd.alcatel-lucent.com endpoint-list 100
```

- 4 Create an Ethernet OAM Maintenance End Point using the **ethoam endpoint admin-state** command. For example:

```
-> ethoam endpoint 100 domain esd.alcatel.com association alcatel-lucent-sales direction up port 1/10
```

- 5 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-lucent-sales admin-state enable
```

- 6 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-lucent-sales ccm enable
```

- 7 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
```

- 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.com association alcatel-lucent-sales
```

Configuring Ethernet OAM

This section describes how to use Alcatel-Lucent OmniSwitch 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
 - 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-lucent.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 primary 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-lucent-sales** in the **esd.alcatel-lucent.com** domain, enter:

```
-> ethoam association alcatel-lucent-sales format string domain esd.alcatel.com  
primary-vlan 10
```

To remove an MA, use the **no** form of this command. For example:

```
-> no ethoam association alcatel-lucent-sales domain esd.alcatel.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-lucent-sales domain esd.alcatel.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-lucent-sales domain esd.alcatel.com ccm-interval  
intervallm
```

To modify the MEP list of an MA, use the **ethoam association endpoint-list** command, as shown:

```
-> ethoam association alcatel-lucent-sales domain esd.alcatel.com endpoint-list  
100-200
```

To remove the MEP list from an Ethernet OAM Maintenance Association, enter:

```
-> no ethoam association alcatel-lucent-sales domain esd.alcatel.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-lucent-sales” Maintenance Association on port 1/2 of primary VLAN 400, enter:

```
-> ethoam end-point 100 domain esd.alcatel.com association alcatel-lucent-sales
direction up port 1/2 primary vlan 400
```

To remove a MEP, use the **no** form of this command. For example:

```
-> no ethoam end-point 100 domain esd.alcatel.com association
alcatel-lucent-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.com association alcatel-lucent-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.com association alcatel-lucent-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.com association alcatel-lucent-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.com association alcatel-lucent-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 from 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.com
association alcatel-lucent_sales flag fdbonly 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.com association
alcatel-lucent_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.com association
alcatel-lucent_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 round-trip 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
association 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 form 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.

CVLANs for EVC MEPs

Configuring Ethernet Virtual Circuits (EVC) MEGs or MEPs on per customer VLAN (CVLAN) in a SVLAN allows supporting connectivity and fault management on per CVLAN basis. The EVC MEG or MEP can be configured on the UNI-N of the provider bridge. The EVC MEG will assist the service provider to instantiate a MEP instance for each customer VLAN on the UNI-N port to perform OAM action for individual CVLAN traffic bound to the EVC.

The following sections provide the details about the CVLAN configuration for the EVC MEPs.

CVLAN Configuration Overview

E-service must be configured prior to configuring MEP on CVLAN.

The CVLAN for the EVC MEP must be configured only on UP MEPs and User-Network (UNI) Port. The CVLAN MEPs associated with the EVC MEG will generate and receive double tagged frames. Maximum of two tags is processed.

The CVLAN associated to the MEP must be part of the allowed CVLAN list configured in the MA for which the MEP belongs.

Creating MEP on CVLAN

MEPs can be created on per CVLAN in a SVLAN. To create a MEP on a CVLAN, use the command `ethoam endpoint` command. For example:

```
-> ethoam endpoint 10 domain MD association MA direction up port 1/1 cvlan 20
```

The direction of the MEP must be UP and the port should be a UNI port using Ethernet service, for which the CVLAN is configured.

Note. The CVLAN ID must be part of the allowed CVLAN list before being configured.

Configuring Remote Fault Propagation (RFP)

Remote Fault propagation (RFP) propagates connectivity fault events into the interface that is attached to a MEP. Once the fault is detected for a MEP, the MEP's interface is shutdown. Unlike other violation mechanisms that keep the link up when an interface is shutdown, this fault propagation mechanism will effectively shutdown the link, so that the remote end of the interface also detects a link down. RFP detects only Loss of connectivity and Remote MAC defect. To enable or disable RFP on MEP, use the command `ethoam endpoint domain association rfp`. For example:

To enable RFP on MEP,

```
-> ethoam endpoint 3 domain mdl association mal rfp enable
```

To disable RFP on MEP,

```
-> ethoam endpoint 3 domain mdl association mal rfp disable
```

Note. Remote Fault Propagation (RFP) is configurable on per MEP basis supported only for UP MEPs. RFP is not supported on virtual UP MEPs.

Configuring the Allowed CVLAN List

The CVLAN ID before being associated to a MEP for monitoring must be configured in the allowed CVLAN list. To configure the allowed CVLAN list use the `ethoam association allowed-cvlan-list` command. For example:

```
-> ethoam association MA domain MD allowed-cvlan-list 10-15
```

Note. The CVLANs configured in the allowed CVLAN list must be associated with the SVLAN of the MA.

Configuring the CVLAN (ctag) Priority

A priority value can be assigned to the outbound packets of the CVLAN. To configure the priority use the **ethoam endpoint ctag-priority** command. For example:

```
-> ethoam endpoint 1 domain mdl association mal ctag-priority 6
```

If the priority value is not configured, by default the outer-tag priority is configured to the inner-tag priority. The priority value range is 0 to 7.

CVLAN Insertion for Untagged Packets

CVLAN insertion for untagged packets is to convert the untagged frames into double tagged frames in the provider network, so as to make ICMP between the endpoints to work. Untagging of the frames should be done on the customer network. Note that when the CVLAN insertion for untagged packets feature is enabled, the legacy behavior of the UNI and NNI port will be disabled. Only one CVLAN can be associated to an UNI port.

To enable or disable the CVLAN insertion for untagged packets, use the command **ethernet-service untagged-cvlan-insert** command. For example:

```
-> ethernet-service untagged-cvlan-insert enable
-> ethernet-service untagged-cvlan-insert disable
```

To associate the CVLAN as untagged to the UNI port, use the command **ethernet-service sap uni untagged-cvlan** command. For example:

```
-> ethernet-service sap 10 uni 1/7 untagged-cvlan 10
```

To configure an SVLAN interface which would map the SVLAN to the CVLAN, use the command **Ip interface cvlan** command. For example:

```
-> ip interface "vlan10" address 10.10.10.1 mask 255.255.255.0 cvlan 10 vlan
1001
```

Note. To view the status of the CVLAN insertion for untagged packets feature, use the command **show ethernet-service untagged-cvlan-insert**. To view CVLAN mapped interface, use the command **show ip interface cvlan**. For more information, see *OmniSwitch AOS Release 6 CLI Reference Guide*.

Viewing the CVLAN Configurations

The following commands displays the CVLAN related information:

| | |
|---|---|
| show ethoam domain association | Displays the CVLANs configured for the MEPs in the MA. |
| show ethoam domain association end-point | Displays the CVLAN configured for the MEP. |
| show ethoam domain | Displays the allowed CVLAN list configured for the MEP. |

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. |

18 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 (for example, 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:

- Configuring SAA for MAC Address on [page 18-4](#).
- Configuring SAA for IP on [page 18-4](#).
- Configuring SAA for Ethoam Loopback on [page 18-4](#).
- Configuring SAA for ETH-DMM on [page 18-4](#).
- Displaying SAA Configuration on [page 18-6](#).

SAA Specifications

The following table lists Ethernet OAM specifications.

| | |
|--------------------------|--|
| IEEE Standards Supported | N/A |
| Platforms Supported | OmniSwitch 6450 Metro license required for OmniSwitch 6450. |
| Minimum SAA interval | 1 minute. |
| Max number of L2 SAA | 128 |

Note. SAA interval value can be configured as 1, 2, 5 or 10 to 1500. Depending on the configuration, it may not be possible to run 128 SAAs. A SAA scheduler resource check is implemented to guarantee the start of a new SAA.

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 |
| Timeout value for Fault Notification Alarm Generation | <code>ethoam fault-alarm-time</code> | 2.5 seconds |
| Timeout value for Fault Notification Generation Reset | <code>ethoam fault-reset-time</code> | 10 seconds |

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 (for example, 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 in the following example:

```
-> 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 "saa1" type ip-ping destination-ip 123.32.45.76 source-ip 123.35.42.124
type-of-service 4
```

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 (default 5min), 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. During ARP resolution packet drop is noticed.

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 "saa1" type ethoam-loopback target-endpoint 10 source endpoint 1 domain
mdl association ma1 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 "saa1" type ethoam-two-way-delay target-endpoint 10 source endpoint 1
domain mdl association ma1 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 following example:

```
-> saa "saal" start
-> saa "saal" stop
```

Enabling Jitter Calculation in SAAs

Jitter is the variation in latency as measured in the variability over time of the packet latency across a network. A network with constant latency has no variation or jitter. In AOS, as per the old design, the inter-arrival jitter calculation is based on the Round Trip Time (RTT) difference between two successive packets. The enhanced mode is to calculate inter-arrival jitter based on the formula specified in RFC 1889.

To set the jitter calculation mode to enhanced mode, use the **saa jitter-calculation enhanced** command.

```
-> saa jitter-calculation enhanced
```

To view the mode which is set for jitter-calculation, use the **show saa config** command.

```
-> show saa config
    Jitter-calculation:enhanced
```

By default the mode will be set to **default** which will calculate the jitter value as per old design.

Displaying the SAA Configuration

To display information about SAA on the switch, use the show commands listed in the table below:

| | |
|-----------------------------|---|
| show saa | Displays generic configuration parameters of all the SAAs maintained at a given time. |
| show saa statistics | Displays SAA statistics. |
| show saa type config | Displays configured SAAs of the given type. |
| show saa config | Displays the mode which is set for jitter calculation. |

19 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 AOS Release 6 CLI Reference Guide*. This chapter provides an overview of LINK OAM and includes the following information:

- [“LINK OAM Specifications” on page 19-2](#)
- [“LINK OAM Defaults” on page 19-3](#)
- [“Quick Steps for Configuring LINK OAM” on page 19-4](#)
- [“Interaction With Other Features” on page 19-7](#)
- [“Configuring Link Monitoring” on page 19-9](#)
- [“Configuring LINK OAM” on page 19-8](#)
- [“Verifying the LINK OAM Configuration” on page 19-11](#)

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 6350, 6450 Metro license required for OmniSwitch 6450. |
| 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 OAM 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 OAMPDU has to 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
```

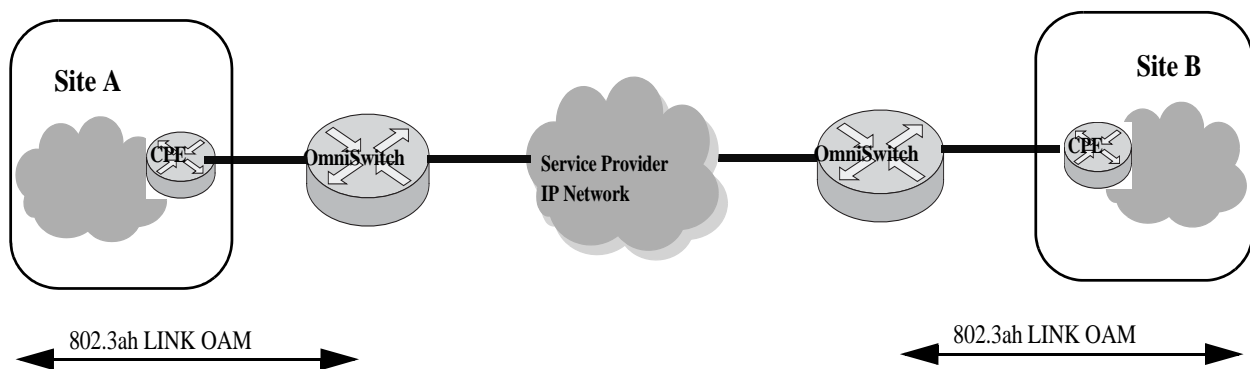

10 Configure the threshold, window and notify-status for errored-frame-seconds-summary on the port by using the **efm-oam errored-frame-seconds-summary** command. For example:

```
-> 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 19-6](#)
- [“Link Monitoring” on page 19-6](#)
- [“Remote Fault detection” on page 19-6](#)
- [“Remote Loopback Testing” on page 19-7](#)

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 19-9](#)

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 19-9](#)

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 19-10](#)

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 does 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** is 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 has to 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 19-6](#) 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 19-6](#)

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 19-7 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 is processed by local LINK OAM port. As a result, remote port is in remote-loopback state and the local port is in 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 is not 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 has to be initiated for loopback session to start.

```
-> efm-oam port 1/1 remote-loopback start
```

The above command initiates 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. |

20 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, and so on. 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include the following:

- Configuring UDLD on [page 20-6](#).
- Configuring the operational mode on [page 20-7](#).
- Configuring the probe-message advertisement timer on [page 20-7](#).
- Configuring the echo-based detection timer on [page 20-7](#).
- Clearing UDLD statistics on [page 20-8](#).
- Recovering a port from UDLD shutdown on [page 20-8](#).
- Displaying UDLD information on [page 20-8](#).

UDLD Specifications

| | |
|---|-----------------------------|
| RFCs supported | Not applicable at this time |
| IEEE Standards supported | Not applicable at this time |
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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, and so on.

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 20-6](#)), configuring the operational mode (see [“Configuring Mode” on page 20-7](#)), configuring and resetting probe-message advertisement timer (see [“Configuring Probe-timer” on page 20-7](#)), configuring and resetting echo-based detection timer (see [“Configuring Echo-wait-timer” on page 20-7](#)), clearing the UDLD statistics on a switch or port (see [“Clearing UDLD Statistics” on page 20-8](#)), and recovering a port from UDLD shutdown (see [“Recovering a Port from UDLD Shutdown” on page 20-8](#)).

Note. See the “UDLD Commands” chapter in the *OmniSwitch AOS Release 6 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 AOS Release 6 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 20-3](#).

21 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.

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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling MAC Retention on [page 21-4](#).
- Detecting a Duplicate MAC Address on [page 21-4](#).
- Configuring MAC Release on [page 21-5](#).

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 |

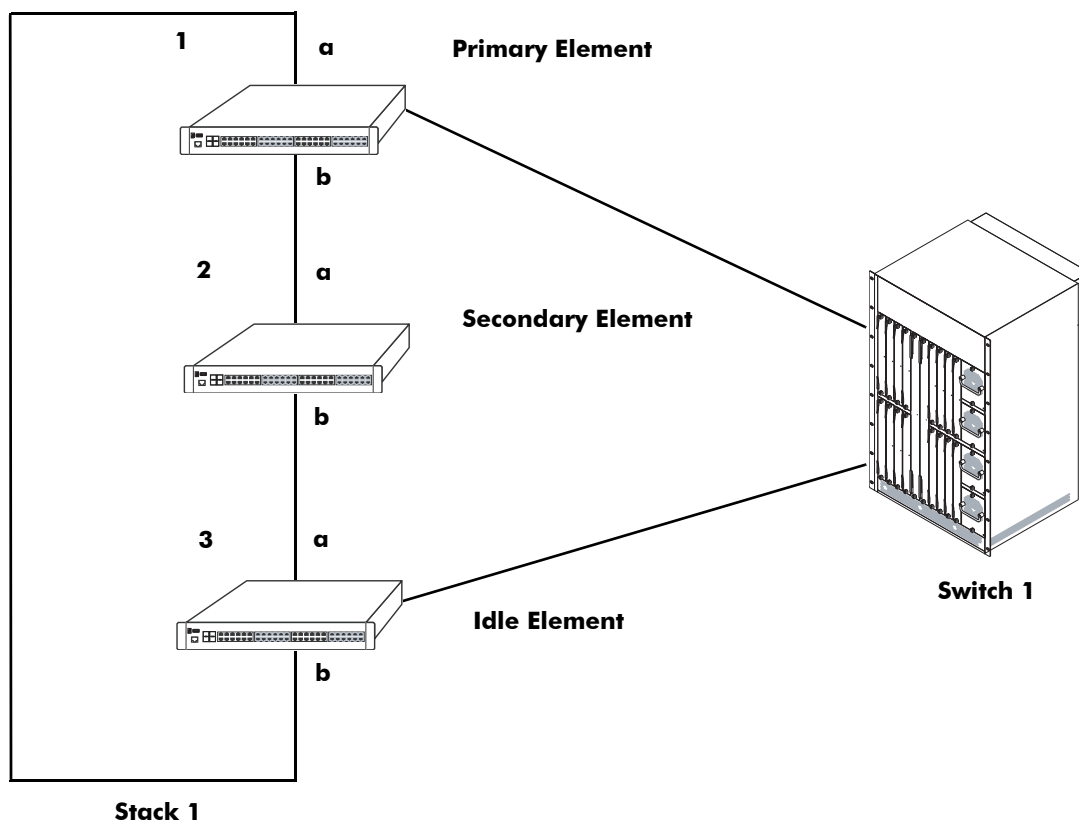
MAC Retention Overview

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'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
```

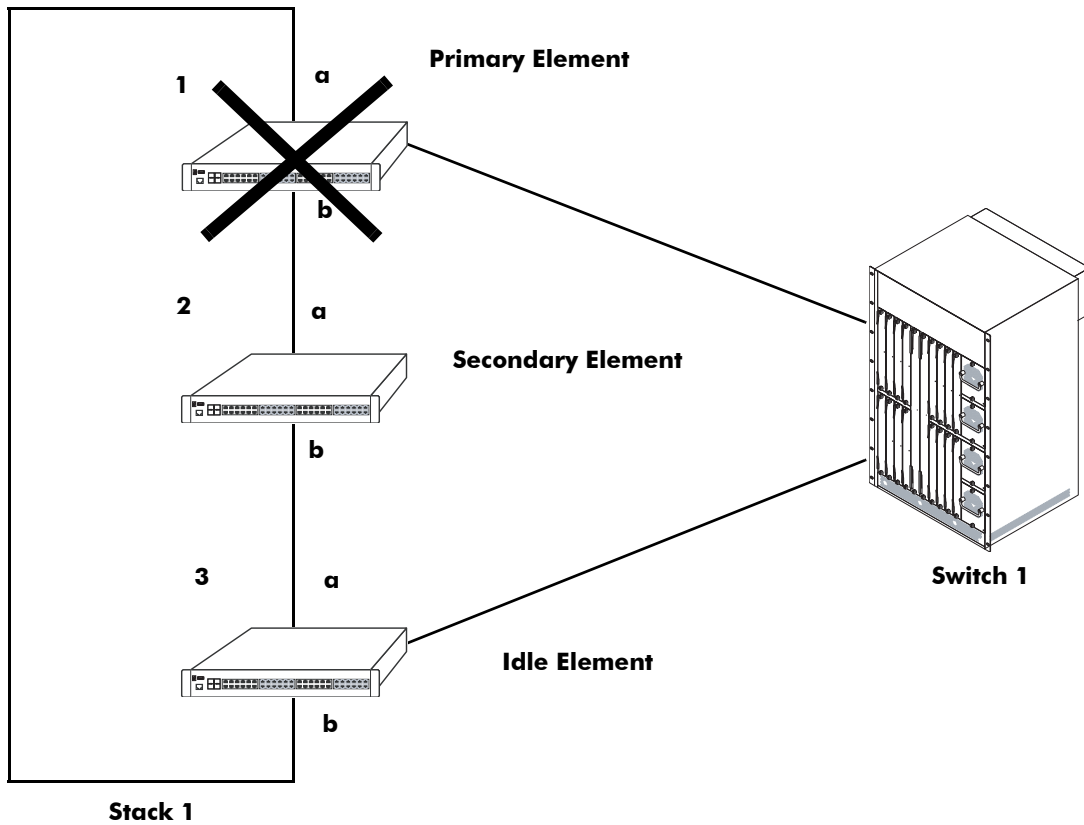
MAC Retention Applications

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.



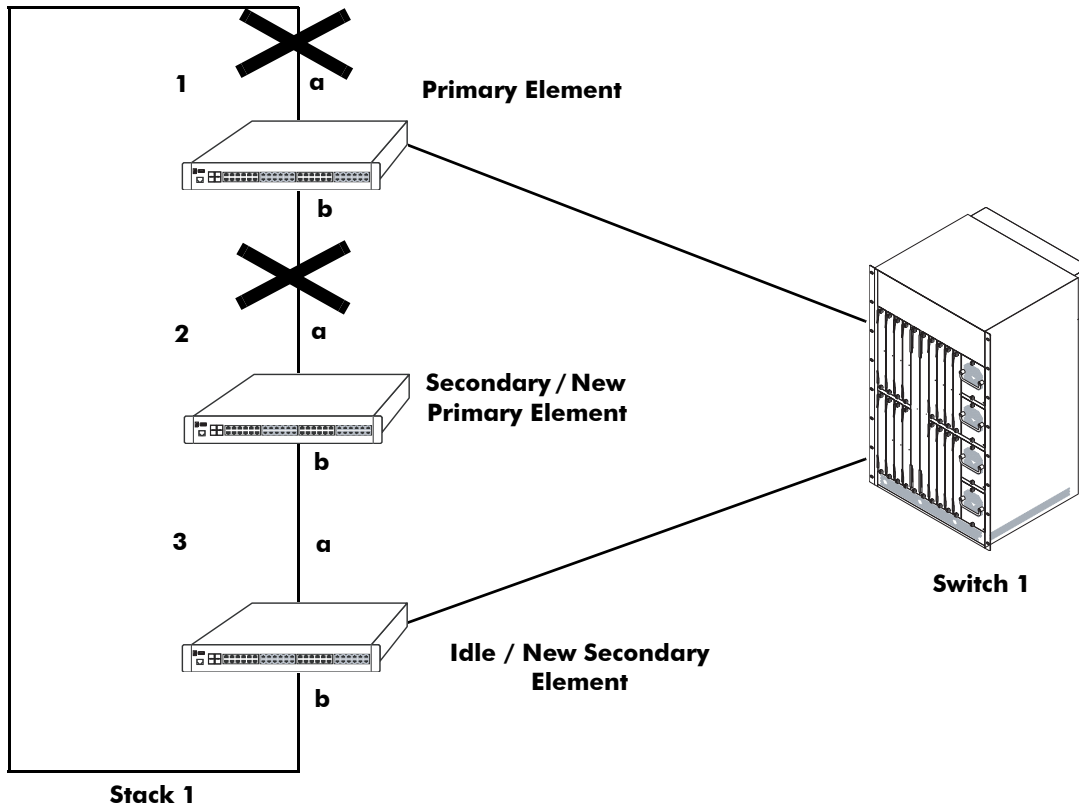
Stack Status when Switch 1 is Down

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.

22 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, the remote device information is periodically updated.

The LLDP Agent Security mechanism can be configured manually for secure access to the network by detecting rogue devices and preventing them from accessing the internal network.

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 13, “802.1AB Commands,”](#) in the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include the following:

- [“Quick Steps for Configuring 802.1AB”](#) on page 22-4
- [“Quick Steps for Configuring LLDP-MED Network Policy”](#) on page 22-5
- [“Configuring LLDPDU Flow”](#) on page 22-16.
- [“Nearest Bridge/Edge Mode”](#) on page 22-14
- [“Enabling and Disabling Notification”](#) on page 22-16.
- [“Enabling and Disabling Management TLV”](#) on page 22-17.
- [“Enabling and Disabling 802.1 TLV”](#) on page 22-17.
- [“Enabling and Disabling 802.3 TLV”](#) on page 22-18.
- [“Enabling and Disabling MED TLV”](#) on page 22-18.
- [“Enabling and Disabling Proprietary TLV”](#) on page 22-19
- [“Setting the Transmit Interval”](#) on page 22-19.
- [“Setting the Transmit Hold Multiplier Value”](#) on page 22-21.
- [“Setting the Transmit Delay”](#) on page 22-21.
- [“Setting the Transmit Fast Start Count”](#) on page 22-21

- [“Setting the Transmit Fast Start Count” on page 22-21.](#)
- [“Setting the Notification Interval” on page 22-21.](#)
- [“Configuring LLDP Security Mechanism” on page 22-22](#)
- [“Verifying 802.1AB Configuration” on page 22-24.](#)

802.1AB Specifications

| | |
|---|--|
| IEEE Specification | <i>IEEE 802.1AB-2005 Station and Media Access Control Connectivity Discovery</i> |
| TIA Specifications | TIA-1057 - Link Layer Discovery Protocol for Media Endpoint Devices |
| Platforms Supported | OmniSwitch 6350, 6450 |
| Transmit time interval for LLDPDUs | 5 to 32768 in seconds |
| Transmit hold multiplier value | 2 to 10 |
| Fast start count | 1 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 transmit interval | 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 |

| Parameter Description | Command | Default Value/Comments |
|-----------------------------------|--|---|
| Reinit delay | lldp reinit delay | 2 seconds |
| Notification interval | lldp notification interval | 5 seconds |
| LLDPDU 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 |
| 802.1 TLV | lldp tlv dot1 | Disable |
| 802.3 TLV | lldp tlv dot3 | Disable |
| LLDP Media Endpoint Device | lldp tlv med | Disable |
| LLDP TLV | lldp tlv proprietary | Disable |
| Mode | lldp destination mac-address | Nearest Bridge |
| LLDP Trust Agent Violation Action | lldp trust-agent violation-action | Trap |

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 transmit interval** 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
```

- 6 Set the LLDPUs transmit fast start count required for LLDP Fast Restart mechanism to be activated.

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),
  Capabilities Supported  = none supported,
  Capabilities Enabled    = none enabled,
```

For more information about this display, see the *OmniSwitch AOS Release 6 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 join 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 | - |
| 2 | guest-voice | - | - | 44 |

To verify the network policies enabled on different slots and ports, use the **show lldp med network-policy** command. For example:

```
-> show lldp med network-policy
```

| slot/port | Network Policy ID |
|-----------|-------------------|
| 1/10 | 1 2 |
| 2/10 | 1 2 |
| 3/10 | 1 2 |

For more information about this display, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

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 learns 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 information contains the MAC service access point (MSAP) identifier and the time period for the validity of the associated information of the LAN device. The following are the mandatory TLVs contained in an LLDPDU:

- 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 the TLVs must be included.

IEEE 802.3 organizationally specific TLV set

- MAC/PHY configuration/status TLV
- Power Via MDI TLV
- 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, and so on). In such a case, OmniSwitch stops sending LLDPDU and starts sending MED LLDPDU on the port connected to the edge device.

LLDP PoE Power Negotiation

The IEEE 802.3 specific TLVs for mac-phy or power-via-mdi can be used for PoE power negotiation.

mac-phy TLV

When **mac-phy** is configured the power class detection is done via hardware by the switch's PoE controller and the maximum power for the port is based on the class of the powered device. Powered devices can draw up to the maximum amount of power allowed for its class without any negotiation with the switch.

power-via-mdi TLV

When power-via-mdi is configured the power for the powered device is negotiated using the optional power via MDI TLV in the LLDPDU. The powered device can request additional power using the power via MDI TLV. The switch will check the current PoE budget and if power is available the switch will provide the requested power to the powered device. If power is unavailable, the switch will respond with the existing maximum power information.

- Power negotiation is supported for Class 4 powered devices.
- The maximum power a powered device can request cannot exceed the maximum power allowed for the PoE class in which the powered device is detected.
- If the port is manually configured with a maximum power value, the powered device cannot receive more power than the maximum configured value.

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.
- Up to eight 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 cannot 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 signaling application must be activated on the OmniSwitch to advertise the VLAN to the connected IP Phones. For example, on how to set up the LLDP-MED for IP Phones, see [“Enabling and Disabling Notification” on page 22-16](#)

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 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 unidirectionally, 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 goes 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 LLDP-specific information of the remote system is stored in the LLDP MIB. The TTL TLV carries a positive value in seconds, and informs the other device as to how long this information is valid. Once a remote device is learned on a local port, the local device discards that entry from its database if the receiving device does not receive an LLDPDU from the same remote device and on the same local port within the TTL mentioned in the previous LLDPDU. This is called the aging time and can be set by the user.

LLDP Agent Security Mechanism

The OmniSwitch LLDP Agent Security mechanism provides a solution for secure access to the network by detecting rogue devices and preventing them from accessing the internal network. LLDP agent security can be achieved by allowing only one trusted LLDP remote agent on a network port.

User is provided an option to configure the Chassis ID subtype that can be used in validating the Chassis ID type in the incoming LLDP PDU. If the Chassis ID is not configured, by default, the first LLDP remote agent is learned with the received Chassis ID. When more than one LLDP agent is learned on a port, the port is moved to a violation state.

For example, when someone tries to take control over the network by connecting non-registered devices to an NNI port, the LLDP Security mechanism is activated. One or both of the following actions are performed according to the security configuration:

- When the rogue device is detected, a violation is reported on the port.
- The NNI port that is connected to the rogue device is blocked. Thus the rogue device is prevented from accessing the internal network.

LLDP security mechanism can be enabled or disabled globally at chassis level, at slot level, or at individual port level. When the LLDP agent security is enabled, the configured ports are monitored for reception of any LLDPDU. When an LLDPDU is received, the remote agent ID is learned and the port is considered as a trusted port if the port does not have any other LLDP remote agent assigned. If the remote agent chassis ID and port IDs received are already present in the trusted remote agent database on the same port, then the port remains in a trusted state.

However, a port is moved to violation state under the following conditions:

- When a link up is received on an LLDP security enabled port, if no LLDPDU is received even after three times the LLDP timer interval period (30 seconds), the port is moved to a violation state.
- If a trusted remote agent exists, and if no LLDP remote agent is learned even after three times the LLDP timer interval period (30 seconds), the port is moved to a violation state.
- If a new LLDP remote agent is learned after the link up and down, then the port is moved to a violation state.
- If the same chassis ID and port ID exist in the trusted remote agent database but on a different port, then the port remote agent is learned and the port is moved to a violation state.
- If a new LLDP remote agent is learned on a port that has a trusted LLDP remote agent, then the port is moved to a violation state.

Three actions can be configured when an LLDP security violation occurs. The different violation actions that can be configured are:

- **trap** - Generate a trap
- **shutdown** - Shutdown the port
- **trap-and-shutdown** - A trap is generated upon shutdown of the port due to violation.

When a shutdown occurs on a port, it can be cleared manually through the CLI interface using the **clear violations** command.

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 chapter [“Managing Automatic Remote Configuration Download”](#) in the *OmniSwitch AOS Release 6 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 Transmission 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 must not be set higher than 30 secs (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 chapter “[Managing Automatic Remote Configuration Download](#)” in the *OmniSwitch AOS Release 6 Switch Management Guide* for a configuration example using the Nearest-Edge Mode with the Automatic-Configuration feature.

Configuring 802.1AB

The following sections list detailed procedures to enable 802.1AB, assign ports, network policies to 802.1AB, and configure the LLDP security mechanism for OmniSwitch.

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 notification 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
```

Enabling and Disabling Proprietary TLV

The OmniSwitch advertises the Access Point location information to the APs connected to it through the Proprietary TLVs. The proprietary TLVs are transmitted along with the LLDP BPDU. After LLDP is configured on the network devices, the NMS can obtain the network topology.

The WLAN management VLAN is transmitted to AP through LLDP using existing Port VLAN TLV. The WLAN VLAN is locally maintained for each port on the switch.

The Proprietary TLV must be enabled to advertise the AP location. The `lldp tlv proprietary` command is used to enable the Proprietary TLV.

To enable the Proprietary TLV transmission on a switch, enter the `lldp tlv proprietary` command, as shown:

```
-> lldp chassis tlv proprietary enable
```

To enable the Proprietary TLV on port, for example port 4 of slot 4, enter the following command at the CLI prompt:

```
-> lldp 4/4 tlv proprietary enable
```

To disable the Proprietary TLV on a switch, enter the `lldp tlv proprietary` command, as shown:

```
-> lldp chassis tlv proprietary disable
```

To disable Proprietary TLV on port, for example port 3 of slot 4, enter the following command at the CLI prompt:

```
-> lldp 4/3 tlv proprietary disable
```

To view the operational status of Proprietary TLV on a switch, use the `show lldp config` command.

```
-> show lldp config
```

```
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
      | Admin | Notify | Std TLV | Mgmt | 802.1 | 802.3 | MED | Proprietary
Slot/Port| Status | Trap | Mask | Address | TLV | Mask | Mask | TLV
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
  3/1  Rx + Tx  Disabled  0x00  Disabled  Disabled  0x00  0x00  Disabled
  3/2  Rx + Tx  Disabled  0x00  Disabled  Disabled  0x00  0x00  Disabled
  3/3  Rx + Tx  Disabled  0x00  Disabled  Disabled  0x00  0x00  Disabled
```

To view the AP location, use the `show lldp local-port` command.

```
-> show lldp local-port
```

```
Local Slot 1/Port 1 LLDP Info:
  Port ID           = 1001 (Locally assigned),
  Port Description  = Alcatel-Lucent 1/1,
  Vlan              = 1,
  AP Location       = sw1,
Local Slot 1/Port 2 LLDP Info:
  Port ID           = 1002 (Locally assigned),
  Port Description  = Alcatel-Lucent 1/2,
  Vlan              = 1,
  AP Location       = -,
```

The AP location and VLAN information can also be viewed from WebView. To view the information, from the WebView page:

- 1 Click on the **Physical** tab.
- 2 Click on **Adjacencies**. Adjacencies home page is displayed.
- 3 In the Adjacencies home page, select **LLDP**.
- 4 Click on **Local** and select **AP Management TLV** from the displayed option. This will display the **AP Management TLV** information such as Slot, Port, VLAN, and AP Location. A sample screen is displayed as follows:

The screenshot displays the 'AP Management TLV' configuration page. The interface includes a navigation sidebar on the left with options like Chassis Mgmt, Health, Ethernet, Console Port, Adjacencies, and WLAN. The main content area shows a table with the following data:

| Slot | Port | Vlan | AP Location |
|------|------|------|-------------|
| 3 | 1 | 777 | PORT1 |
| 3 | 2 | 1 | - |
| 3 | 4 | 777 | - |
| 3 | 5 | 777 | - |
| 3 | 6 | 777 | - |
| 3 | 7 | 777 | - |
| 3 | 8 | 777 | - |
| 3 | 9 | 777 | - |
| 3 | 10 | 777 | - |
| 3 | 11 | 777 | - |
| 3 | 12 | 777 | - |
| 3 | 13 | 777 | - |
| 3 | 14 | 777 | - |
| 3 | 15 | 777 | - |
| 3 | 16 | 777 | - |
| 3 | 17 | 777 | - |
| 3 | 18 | 777 | - |
| 3 | 19 | 777 | - |
| 3 | 20 | 777 | - |
| 3 | 21 | 777 | - |
| 3 | 22 | 777 | - |
| 3 | 23 | 1 | - |
| 3 | 24 | 777 | - |
| 3 | 25 | 777 | - |
| 3 | 26 | 777 | - |

At the bottom of the table, there are 'Refresh' and 'Help' buttons.

Note. For more information about WebView, see the *OmniSwitch AOS Release 6 Switch Management Guide*.

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 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 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.

Configuring LLDP Security Mechanism

The **lldp trust-agent** command is used to enable or disable the LLDP security mechanism globally at chassis level, for a slot, or an individual port.

To enable LLDP trust agent globally at chassis level, enter the **lldp trust-agent** command as shown:

```
-> lldp chassis trust-agent enable
```

To enable LLDP trust agent at slot number 1, enter the command as shown:

```
-> lldp 1 trust-agent enable
```

To enable LLDP trust agent at individual port 3 of slot 1, enter the command as shown:

```
-> lldp 1/3 trust-agent enable
```

The chassis ID subtype is configured to validate the remote agent as a trust agent. To set the **chassis-id-subtype** for the LLDP trust agent globally at chassis level as **chassis-component**, enter the **lldp trust-agent** command as shown:

```
-> lldp chassis trust-agent chassis-id-subtype chassis-component
```

To set the **chassis-id-subtype** for the LLDP trust agent on the individual port 3 of slot 1 as **port-component**, enter the **lldp trust-agent** command as shown:

```
-> lldp 1/3 trust-agent chassis-id-subtype port-component
```

Note. By default, the first remote agent with any chassis ID sub type is accepted as a trust agent, if no **chassis-id-subtype** component is specified to validate the remote agent.

To set the action to be performed when a violation is detected globally at the chassis level, use the **lldp trust-agent violation-action** command as shown:

```
-> lldp chassis trust-agent violation-action trap-and-shutdown
```

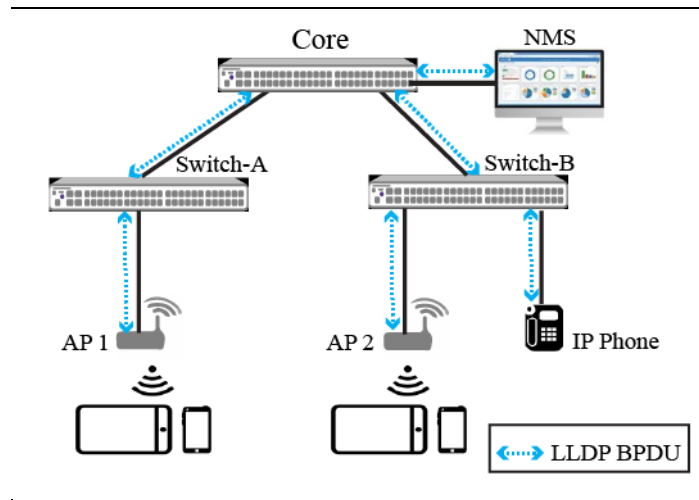
To set the action to be performed when a violation is detected at the individual slot level, use the **lldp trust-agent violation-action** command as shown:

```
-> lldp 1 trust-agent violation-action shutdown
```

Note. For further details on verifying LLDP configuration and trust agent information, see [“Verifying 802.1AB Configuration” on page 22-24.](#)

Application Example - LLDP MED

The following example describes how to configure LLDP MED on the devices.



Application Example - LLDP MED

In the above example, the NMS obtains Layer 2 information about Core Switch, SwitchA, SwitchB, and AP. By using the Layer 2 information, a network administrator can know the detailed network topology information and configuration conflicts. These requirements can be met by configuring LLDP on SwitchA and SwitchB. In addition, the administrator requires that SwitchA and SwitchB send LLDP traps to the NMS, when the LLDP management address changes, global LLDP is enabled or disabled.

For more information on the configuration procedure, see [“Configuring 802.1AB” on page 22-16](#).

Verifying 802.1AB Configuration

To display information about the ports configured to handle 802.1AB, use the following show command:

| | |
|---|---|
| show lldp system-statistics | Displays system-wide statistics. |
| 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. |
| show lldp trust-agent | Displays information of the local LLDP agent or port. |
| show lldp trusted remote-agent | Displays information on trusted remote-agents. |

Note.

The **show lldp trust-agent** command is used to verify the LLDP security configuration. When LLDP security is disabled, the **show lldp trust-agent** command displays the **Admin Status** as **Disabled** for all the ports. However, default values are displayed for the output fields - **Violation Action** as **Trap only**, the **Violation Status** as **Trusted**, and **Chassis ID Subtype** as **8 (Any)**.

Example

```
-> lldp chassis trust-agent disable
-> show lldp 1/1 trust-agent
```

| Slot/Port | Admin Status | Violation Action | Violation Status | ChassisSubtype |
|-----------|--------------|------------------|------------------|----------------|
| 1/1 | Disabled | Trap Only | Trusted | 8(Any) |

For more information about the resulting display, see [Chapter 13, “802.1AB Commands,”](#) in the *OmniSwitch AOS Release 6 CLI Reference Guide*.

23 Using Interswitch Protocols

Alcatel Interswitch Protocol (AIP) is used to discover adjacent switches in the network. The following protocol is supported:

Alcatel 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 23-3](#).

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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Activating AMAP on [page 23-5](#).
- Configuring the AMAP discovery time-out interval on [page 23-5](#).
- Configuring the AMAP common time-out interval on [page 23-6](#).

For information about statically and dynamically assigning switch ports to VLANs, see [Chapter 7, “Assigning Ports to VLANs.”](#)

For information about defining VLAN rules that allow dynamic assignment of mobile ports to a VLAN, see [Chapter 9, “Defining VLAN Rules.”](#)

AIP Specifications

| | |
|---|---|
| Standards | Not applicable at this time. AMAP is an Alcatel proprietary protocol. |
| Platforms Supported | OmniSwitch 6350, 6450 |
| Maximum number of IP addresses propagated by AMAP | 255 |

AMAP Defaults

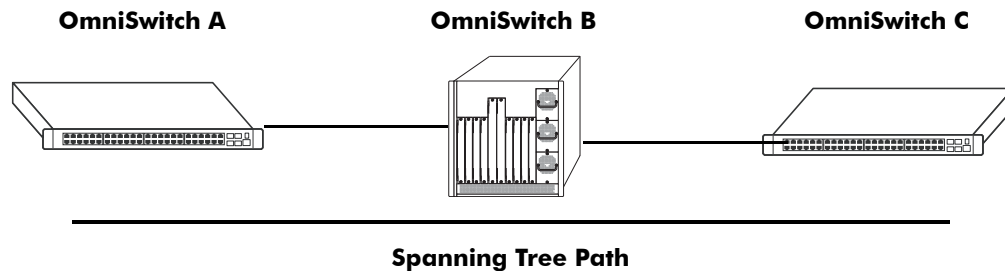
| Parameter Description | Command | Default |
|-------------------------|--|-------------|
| AMAP status | <code>amap</code> | Enabled |
| Discovery time interval | <code>amap discovery time</code> | 30 seconds |
| Common time interval | <code>amap common time</code> | 300 seconds |

AMAP Overview

The Alcatel 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.

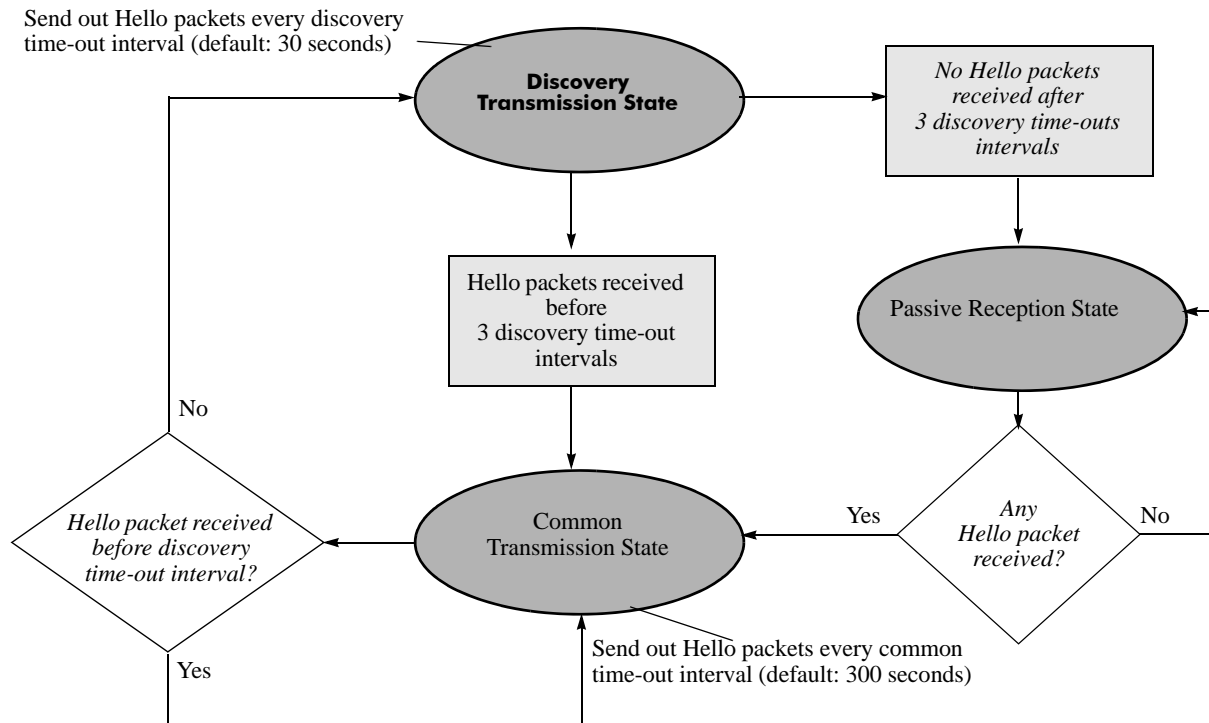


AMAP Transmission States

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 23-3](#).



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

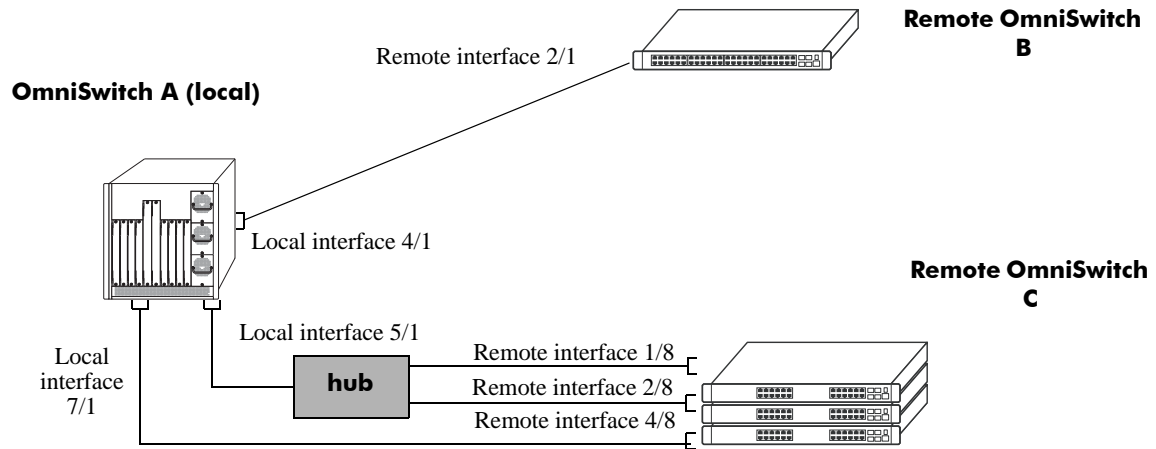
Remote Host 'OmniSwitch B' On Port 4/1 Vlan 1:
Remote Host Device      = OS6450-48SF,
Remote Base MAC        = 00:20:xx:xx:xx:40,
Remote Interface       = 2/1,
Remote VLAN            = 1,
Number of Remote IP Address(es) Configured = 4,
Remote IP(s) =
18.1.1.1
27.0.0.2
192.168.10.1
192.206.184.40

Remote Host 'OmniSwitch C' On Port 5/1 Vlan 7:
Remote Device          = OS6450,
Remote Base MAC        = 00:20:xx:xx:xx: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          = OS6450,
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          = OS6450,
Remote Base MAC        = 00:20:xx:xx:xx: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 AOS Release 6 CLI Reference Guide* for information about the **show amap** command.

24 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 AOS Release 6 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 24-5](#).
- Setting up an 802.1Q VLAN for a link aggregation group. See [“Enabling Tagging with Link Aggregation” on page 24-5](#).
- Configuring 802.1Q VLAN parameters. See [“Configuring the Frame Type” on page 24-6](#).

For information on creating and managing VLANs, see [Chapter 4, “Configuring VLANs.”](#)

For information on creating and managing link aggregation groups, see [Chapter 25, “Configuring Static Link Aggregation”](#) and [Chapter 26, “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 6350, 6450 |
| 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 6350, 6450. |

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 7, “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'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.

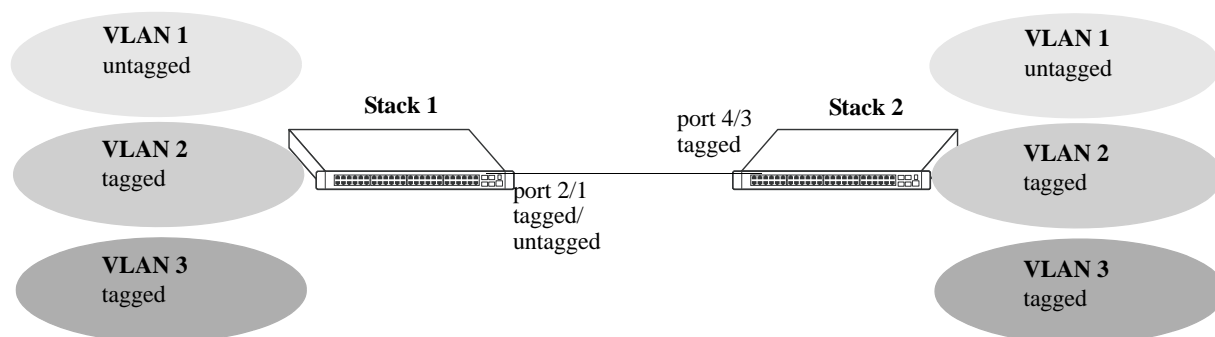
On the ingress side, packets are classified in a VLAN. After classifying a packet, the switch adds an 802.1Q header to the packet. Egress processing of packets is done by the switch hardware. Packets have an 802.1Q tag, which may be stripped off based on 802.1Q tagging/stripping rules.

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 39, “Configuring QoS.”](#)

Alcatel’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 AOS Release 6 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 AOS Release 6 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 25, “Configuring Static Link Aggregation,”](#) or [Chapter 26, “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 AOS Release 6 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 AOS Release 6 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
```

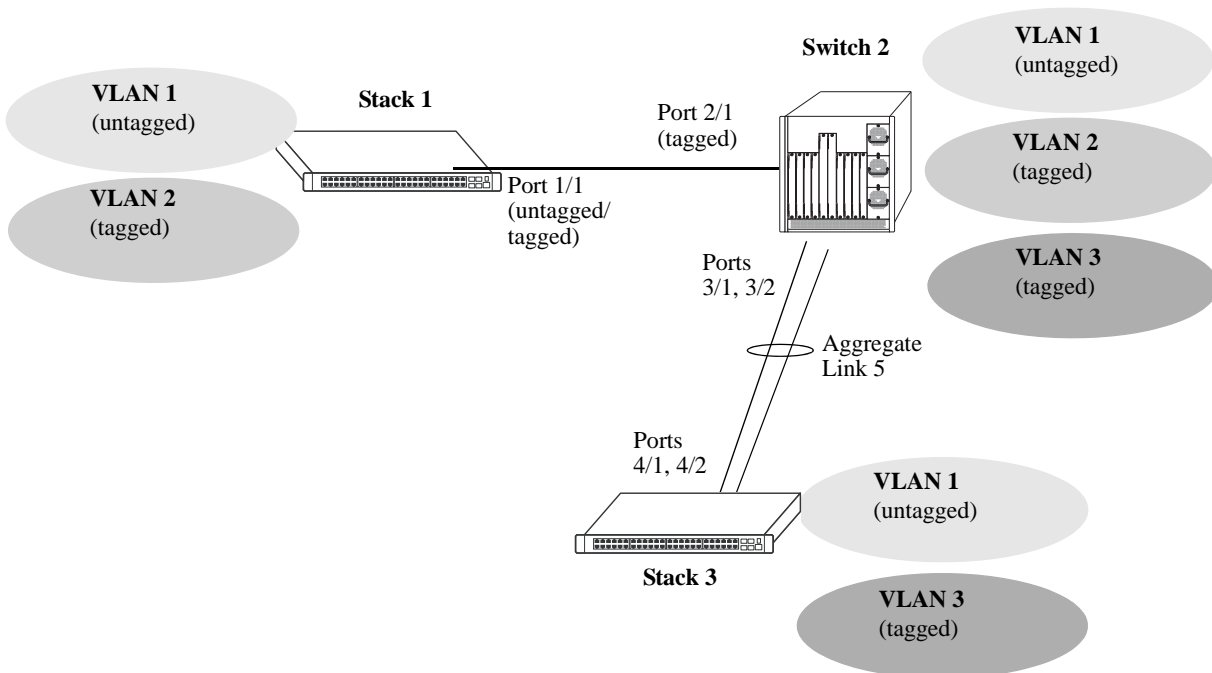
To display all VLANs, enter the following command:

```
-> 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 15, “802.1Q Commands,”](#) in the *OmniSwitch AOS Release 6 CLI Reference Guide*.

25 Configuring Static Link Aggregation

Alcatel 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 (128 groups on chassis-based switches) 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Configuring static link aggregation groups on [page 25-7](#).
- Adding and deleting ports from a static aggregate group on [page 25-9](#).
- Modifying static link aggregation default values on [page 25-10](#).

Note. You can also configure and monitor static link aggregation with WebView, Alcatel 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. Refer to WebView 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 6350, 6450 |
| Maximum number of link aggregation groups | 32 for a standalone switch or a stack of switches 128 for a chassis-based switch |
| Link aggregate group size (number of links that can be configured per link aggregate group) | 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 AOS Release 6 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 25-12 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 for a standalone switch or a stack of switches and up to 128 groups for a chassis-based switch. 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 software treats these virtual links just like physical links. (See [“Relationship to Other Features”](#) on page 25-6 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 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 26, “Configuring Dynamic Link Aggregation.”](#)

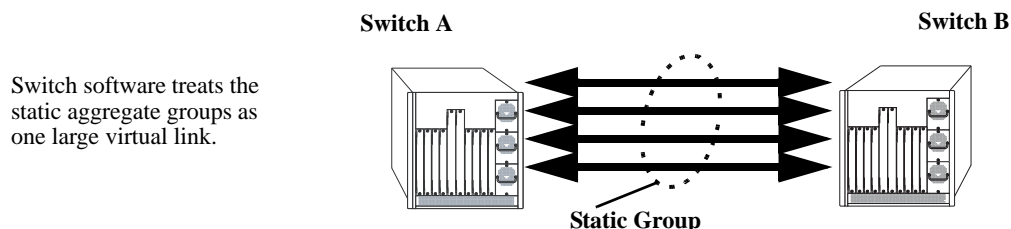
Static Link Aggregation Operation

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 for a standalone switch or a stack of switches and 128 groups for a chassis-based switch.

Static aggregate groups can be created between all OmniSwitch products:

Note. Static aggregate groups cannot be created between an OmniSwitch 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 25-7 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 25-12 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 24, “Configuring 802.1Q.”](#)
- **Spanning Tree.** For more information on Spanning Tree see [Chapter 25, “Configuring Static Link Aggregation.”](#)

Note. See [“Application Example” on page 25-11](#) for tutorials on using link aggregation with other features.

Configuring Static Link Aggregation Groups

This section describes how to use Alcatel Command Line Interface (CLI) commands to configure static link aggregate groups. See [“Configuring Mandatory Static Link Aggregate Parameters” on page 25-7](#) for more information.

Note. See [“Quick Steps for Configuring Static Link Aggregation” on page 25-3](#) for a brief tutorial on configuring these mandatory parameters.

Alcatel 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 25-2](#). If you need to modify any of these parameters, please see [“Modifying Static Aggregation Group Parameters” on page 25-10](#) for more information.

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch AOS Release 6 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 25-8](#).
- 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 25-9](#).

Note. Depending on the needs of your network you can need to configure additional parameters. Commands to configure optional static aggregate parameters are described in [“Modifying Static Aggregation Group Parameters” on page 25-10](#).

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 for a standalone switch or a stack of switches and up to 128 groups for a chassis-based switch. 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 must 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 must 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 (for example, “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 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 (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 can belong to only one aggregate group. In addition, mobile ports cannot be aggregated. See [Chapter 7, “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 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 configuration. See [Chapter 25, “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 25-10)
- Static aggregate group administrative state (see “[Modifying the Static Aggregate Group Administrative State](#)” on page 25-10)

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 (for example, “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

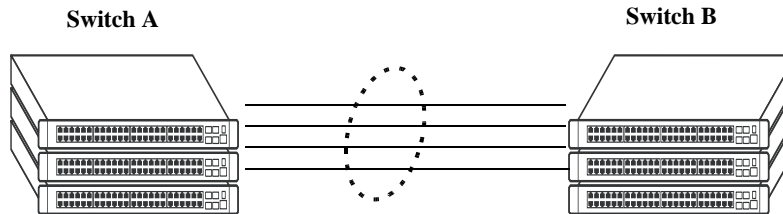
To disable a static aggregate group by entering **static linkagg** followed by the number of the group and **admin state disable**. For example, to disable static aggregate group 1 you would enter:

```
-> static linkagg 1 admin state disable
```

Application Example

Static link aggregation groups are treated by the switch 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 (Switch A) switch are provided here since the steps to configure the remote (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 AOS Release 6 CLI Reference Guide* for complete documentation of **show** commands for link aggregation.

26 Configuring Dynamic Link Aggregation

Alcatel 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 (128 groups on chassis-based switches) 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Configuring dynamic link aggregation groups on [page 26-9](#).
- Configuring ports so they can be aggregated in dynamic link aggregation groups on [page 26-11](#).
- Modifying dynamic link aggregation parameters on [page 26-13](#).
- Configuring Dual-Home Link (Active-Active) on [page 26-29](#).
- Configuring Dual-Home Link (Active-Standby) on [page 26-29](#).

Note. You can also configure and monitor dynamic link aggregation with WebView, Alcatel 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 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 6350, 6450 |
| Maximum number of link aggregation groups | 32 for a standalone switch or a stack of switches 128 for a chassis-based switch |
| Range for optional group name | 1 to 255 characters |
| Link aggregate group size (number of links that can be configured per link aggregate group) | 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 AOS Release 6 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 |
| Wait to Restore Timer | lacp linkagg wait-to-restore-timer | 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 26-33](#) 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 for a standalone switch or a stack of switches and up to 128 groups for a chassis-based switch. 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 26-8](#) 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 link aggregation software allows you to configure the following two different types of link aggregation groups:

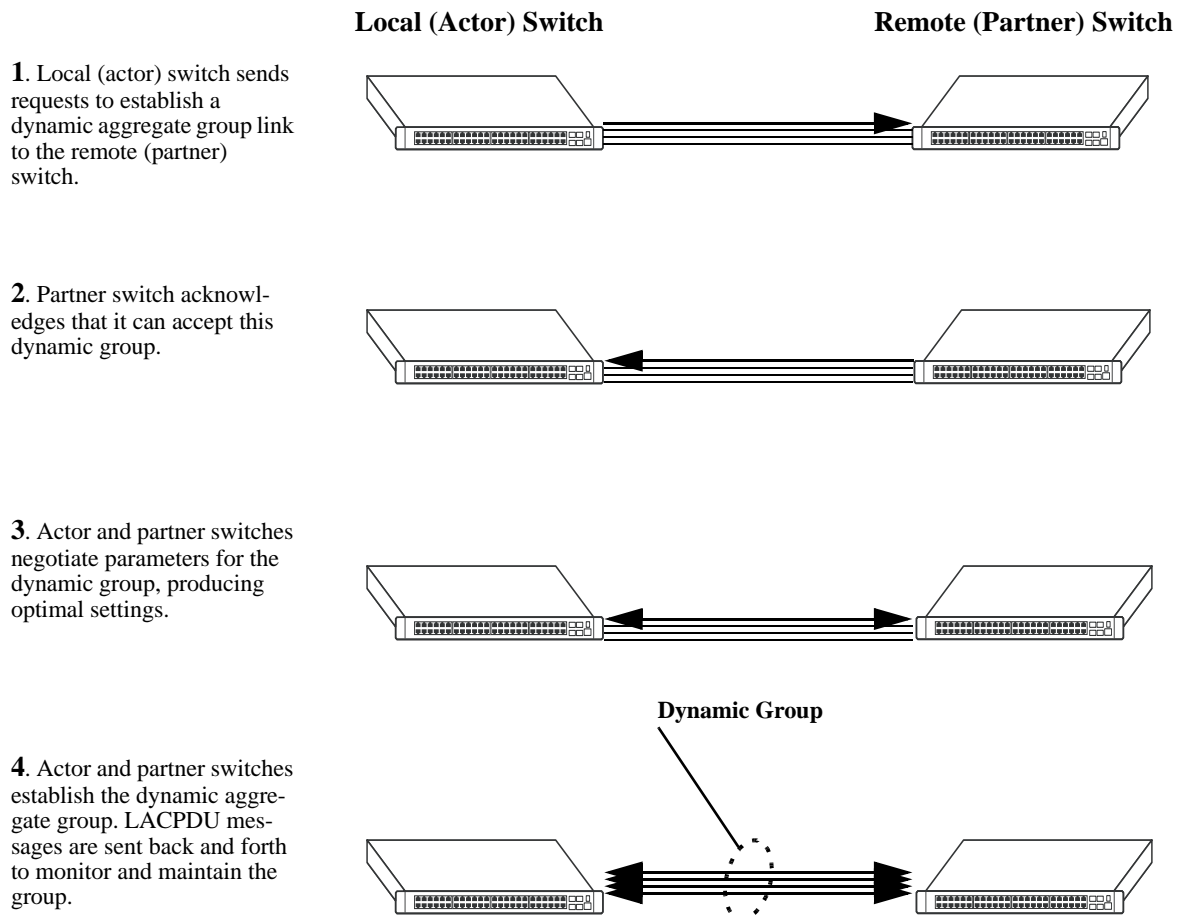
- Static link aggregate groups
- Dynamic link aggregate groups

This chapter describes dynamic link aggregation. For information on static link aggregation, please refer to [Chapter 9, “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 switches.
- an OmniSwitch 6350, 6450 switch and an early-generation Alcatel-Lucent switch.
- an OmniSwitch 6350, 6450 switch and another vendor switch if that vendor supports IEEE 802.3ad LACP.

See [“Configuring Dynamic Link Aggregate Groups” on page 26-9](#) 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 26-33](#) 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 6, “Configuring 802.1Q.”](#)
- **Spanning Tree.** For more information on Spanning Tree, see [Chapter 5, “Configuring Spanning Tree Parameters.”](#)
- **Edge Feature - LACP WTR Delay on Bootup.** For more information on WTR timer, see [“Edge Feature - LACP WTR Delay on Bootup” on page 10-28](#)

Note. See [“Application Examples” on page 26-29](#) for tutorials on using link aggregation with other features.

Configuring Dynamic Link Aggregate Groups

This section describes how to use Alcatel Command Line Interface (CLI) commands to create, modify, and delete dynamic aggregate groups. See [“Configuring Mandatory Dynamic Link Aggregate Parameters” on page 26-9](#) for more information.

Note. See [“Quick Steps for Configuring Dynamic Link Aggregation” on page 26-4](#) for a brief tutorial on configuring these mandatory parameters.

Alcatel 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 26-3](#). For most configurations, using only the steps described in [“Creating and Deleting a Dynamic Aggregate Group” on page 26-10](#) 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 26-3](#), please see [“Modifying Dynamic Link Aggregate Group Parameters” on page 26-13](#) for more information.

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch AOS Release 6 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 26-10](#).
- 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 26-11](#).

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 26-13](#). 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 for a standalone switch or a stack of switches and up to 128 groups for a chassis-based switch. 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 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 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 26-14.

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 (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 5, “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 26-18 and [“Modifying the Partner Port System Administrative State”](#) on page 26-22, 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 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 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 26-3](#) lists default group and port settings for Alcatel 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 26-13](#)
- Dynamic aggregate actor ports beginning on [page 26-18](#)
- Dynamic aggregate partner ports beginning on [page 26-22](#)

Note. You *must* create a dynamic aggregate group before you can modify group or port parameters. See [“Configuring Dynamic Link Aggregate Groups” on page 26-9](#) 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 26-14](#))
- Group administrative state (see [“Modifying the Dynamic Aggregate Group Administrative State” on page 26-14](#))
- Group local (actor) switch actor administrative key (see [“Configuring and Deleting the Dynamic Aggregate Group Actor Administrative Key” on page 26-15](#))
- Group local (actor) switch system priority (see [“Modifying the Dynamic Aggregate Group Actor System Priority” on page 26-15](#))
- Group local (actor) switch system ID (see [“Modifying the Dynamic Aggregate Group Actor System ID” on page 26-16](#))
- Group remote (partner) administrative key (see [“Modifying the Dynamic Aggregate Group Partner Administrative Key” on page 26-16](#))
- Group remote (partner) system priority (see [“Modifying the Dynamic Aggregate Group Partner System Priority” on page 26-17](#))
- Group remote (partner) switch system ID (see [“Modifying the Dynamic Aggregate Group Partner System ID” on page 26-17](#))

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 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 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 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 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 (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 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 (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 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 (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 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 26-18)
- Actor port system ID (see [“Modifying the Actor Port System ID”](#) on page 26-20)
- Actor port system priority (see [“Modifying the Actor Port System Priority”](#) on page 26-20)
- Actor port priority (see [“Modifying the Actor Port Priority”](#) on page 26-21)

Note. See [“Configuring Ports to Join and Removing Ports in a Dynamic Aggregate Group”](#) on page 26-11 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 CLI syntax. However, these keywords do not modify a port 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 5, “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 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. |

| lACP agg actor admin state Keyword | Definition |
|---|---|
| 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. |
| 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 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 (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 (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 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 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 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 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 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 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 26-22](#))
- Partner port administrative key (see [“Modifying the Partner Port Administrative Key” on page 26-24](#))
- Partner port system ID (see [“Modifying the Partner Port System ID” on page 26-24](#))
- Partner port system priority (see [“Modifying the Partner Port System Priority” on page 26-25](#))
- Partner port administrative state (see [“Modifying the Partner Port Administrative Status” on page 26-26](#))
- Partner port priority (see [“Modifying the Partner Port Priority” on page 26-26](#))

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 CLI syntax. However, these keywords do not modify a port 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 5, “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 (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 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 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 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 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 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 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 (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 (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 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 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 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 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 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 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 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 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
```

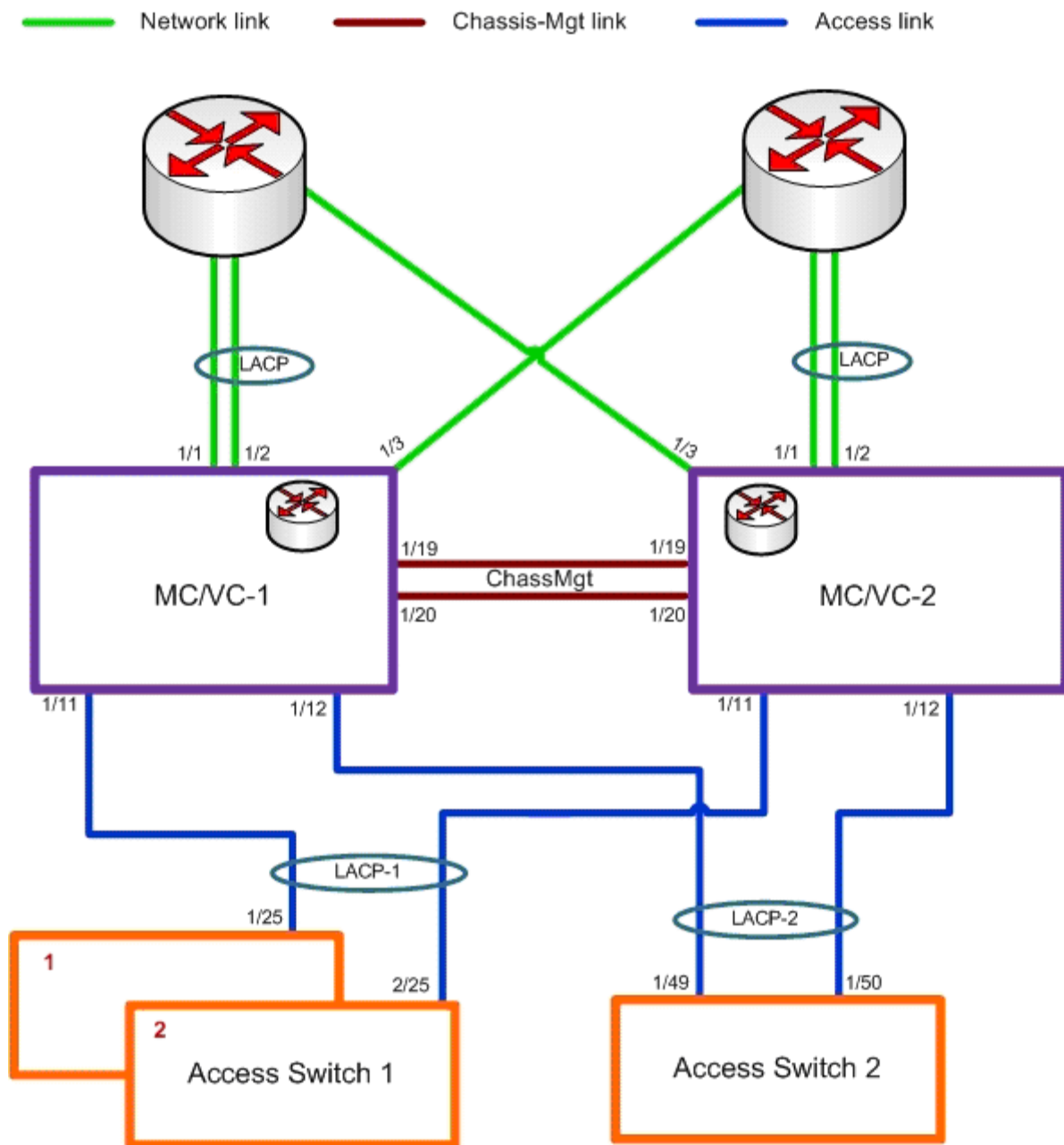
Edge Feature - LACP WTR Delay on Bootup

The LACP WTR delay on bootup is applied on access switches connected to upstream Multi-Chassis or Virtual-Chassis devices using multiple links. It improves convergence when an upstream chassis that went down comes back up. By delaying the restoration of the LACP link long enough to allow L3 to converge, the traffic loss is kept to a minimum. As an added benefit, if an LACP link starts flapping, no traffic will be sent through that link until it is stable (until it is up longer than the WTR timer).

When a chassis which is part of Multi-Chassis or Virtual-Chassis powers up, the VFL links come up immediately and all other links come up after a configured delay (usually 45 seconds). The non VFL links include Network links which connect to upstream switches/routers and Access links which connect to L2 Access switches.

The LACP sync-up within milliseconds after the links come up and traffic originating from the Access switches re-hash and are re-sent to the recovering upstream chassis. At that time, L3 protocols on the MC/VC chassis is not up yet and traffic is redirected to the other MC/VC chassis (or black-holed). Later, after the L3 protocols converge, the traffic is re-routed using the new best routes. This causes a reconvergence double-hit which may exceed 1 second.

Without this feature, the LACP links sync up within milliseconds after the links come up and traffic originating from the access switches re-hash and are re-sent to the recovering upstream chassis. At that time, L3 protocols on the MC/VC chassis is not up yet and traffic is redirected to the other MC/VC chassis (or black-holed). Later, after the L3 protocols converge, the traffic is re-routed using the new best routes. This causes a reconvergence double-hit which may exceed 1 second.



Perform the following procedure to enable WTR timer to edge switches that are unaware that they are connected to multi-chassis so it applies to regular links.

When a LACP comes up, it is required check if there is a WTR enabled for the linkagg.

If there is no WTR enabled for this linkagg, bring up the link. If there is a WTR configured for this linkagg, do the following:

- 1 If there are no other links attached to the same linkagg, bypass the WTR and bring up the link immediately.
- 2 If there are links attached to the same linkagg, start the WTR.

- 3 When the WTR expires, bring up the link.

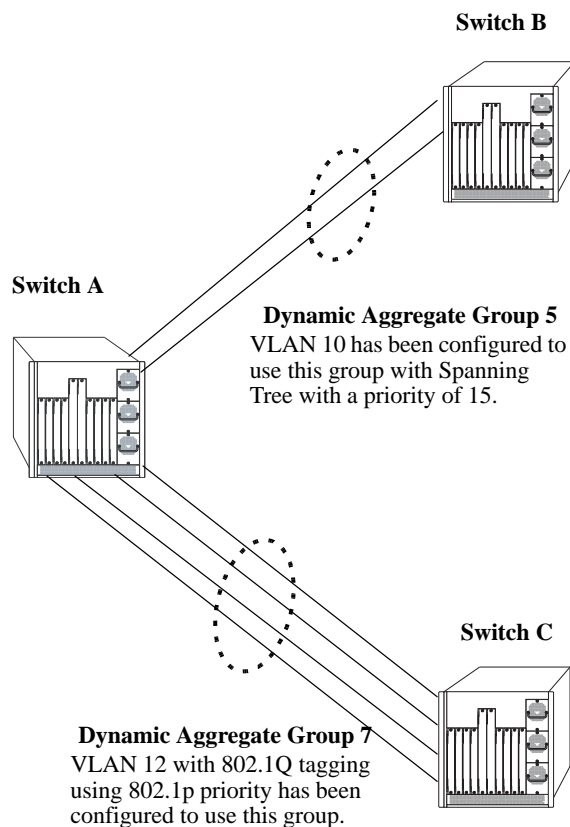
See “[lacp linkagg wait-to-restore-timer](#)” on page 4-233 in Chapter 4, “Link Aggregation Commands” to configure the **wait-to-restore timer**.

Application Examples

Dynamic link aggregation groups are treated by the switch 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.

Dynamic Link Aggregation Example

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 26-30. The steps to configure VLAN 12 (802.1Q and 802.1p example) are described in “[Link Aggregation and QoS Example](#)” on page 26-31.

- Repeat steps 1 through 5 on Switch B. All the commands would be the same except you would substitute the appropriate port numbers.

Link Aggregation and QoS Example

As shown in the figure on [page 26-29](#), 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 (Switch A) switch are provided here since the steps to configure the remote (Switch C) switch would not be significantly different.

- Configure dynamic aggregate group 7 by entering:

```
-> lacp linkagg 7 size 4
```

- 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
```

- Create VLAN 12 by entering:

```
-> vlan 12
```

- Configure 802.1Q tagging with a tagging ID (VLAN ID) of 12 on dynamic aggregate group 7 by entering:

```
-> vlan 12 802.1q 7
```

- 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.

- Configure a policy condition for VLAN 12 called “vlan12_condition” by entering:

```
-> policy condition vlan12_condition destination vlan 12
```

- Configure an 802.1p policy action with the highest priority possible (7) for VLAN 12 called “vlan12_action” by entering:

```
-> policy action vlan12_action 802.1p 7
```

- 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
```

10 Repeat steps 1 through 9 on Switch C. All the commands would be the same except you would substitute the appropriate port numbers.

Note. If you do not use the **qos apply** command any QoS policies you configured will be lost on the next switch reboot.

Displaying Dynamic 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 dynamic and static) 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 the link aggregate group and 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 2 that is attached to dynamic link aggregate group 1 you would enter:

```
-> show linkagg port 2/1
```

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 AOS Release 6 CLI Reference Guide* for complete documentation of **show** commands for link aggregation.

27 Configuring Dual-Home Links

Dual-Home Link (DHL) is a high availability feature that provides fast failover between core and edge switches without implementing Spanning Tree.

DHL Active-Active—an edge technology that splits a number of VLANs between two active links. The forwarding status of each VLAN is modified by DHL to prevent network loops and maintain connectivity to the core when one of the links fails. This solution does not require link aggregation.

In This Chapter

This chapter describes the basic components of DHL solutions 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 AOS Release 6 CLI Reference Guide*.

Information and procedures described in this chapter include:

- [“Dual-Home Link Aggregation Specifications” on page 27-3.](#)
- [“Dual-Home Link Active-Active Defaults” on page 27-4](#)
- [“Dual-Home Link Active-Active” on page 27-5.](#)
- [“Configuring DHL Active-Active” on page 27-8.](#)
- [“Dual-Home Link Active-Active Example” on page 27-10.](#)
- [“Displaying the Dual-Home Link Configuration” on page 27-14.](#)

Dual-Home Link Aggregation Specifications

The table below lists specifications for dynamic aggregation groups and ports:

| | |
|---------------------------------------|---|
| IEEE Specifications Supported | IEEE Std 802.1D, Media Access Control (MAC) Bridges |
| Platforms Supported | OmniSwitch 6350, 6450 |
| DHL session supported | 1 per switch |
| Maximum number of vlans per DHL group | 128 |
| Number of links per group supported | 2, 4, or 8 |

Dual-Home Link Active-Active Defaults

The table below lists default values for dual-home link aggregate groups.

| Parameter Description | Command | Default Value/Comments |
|---------------------------------------|---------------------------------|--|
| DHL session ID | dhl num | If a name is not assigned to a dhl session, the session is configured as DHL-1 |
| Admin state of dhl session | dhl num admin-state | disable |
| Configure a port/link agg as DHL | dhl num linka linkb | NA |
| Configure a VLAN-MAP | dhl num vlan-map linkb | NA |
| Pre-emption timer for the DHL session | dhl num pre-emption-time | 30 seconds |

Dual-Home Link Active-Active

Dual-Home Link (DHL) Active-Active is a high availability feature that provides fast failover between core and edge switches without using Spanning Tree. To provide this functionality, DHL Active-Active splits a number of VLANs between two active links. The forwarding status of each VLAN is modified by DHL to prevent network loops and maintain connectivity to the core when one of the links fails.

This implementation of DHL Active-Active is provided in addition to the previously released LACP-based DHL Active-Standby solution (see [“Dual-Home Link Active-Active Example”](#) on page 27-10). The DHL Active-Active feature is configurable on regular switch ports and on logical link aggregate ports (linkagg ID) instead of just LACP aggregated ports. In addition, the two DHL links are both active, as opposed to the active and standby mode used with LACP.

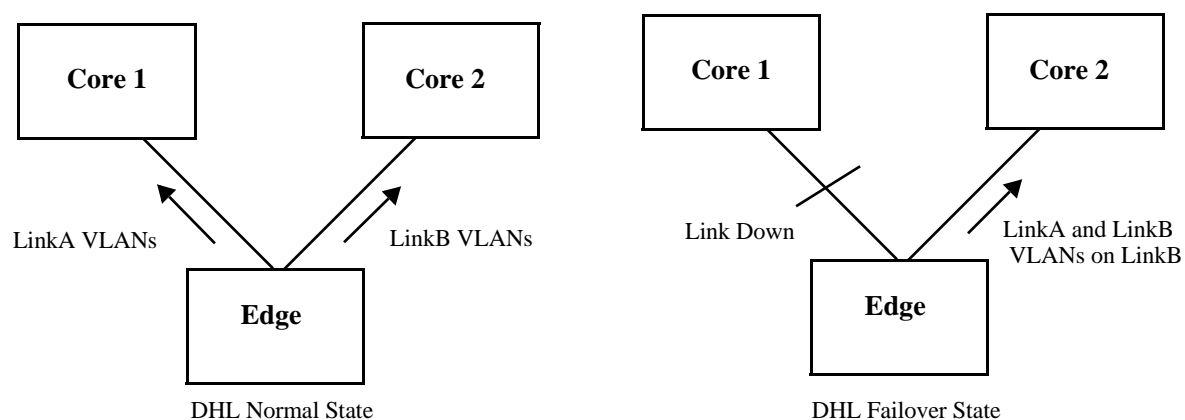
DHL Active-Active Operation

A DHL Active-Active configuration consists of the following components:

- A DHL session. Only one session per switch is allowed.
- Two DHL links associated with the session (link A and link B). A physical switch port or a logical link aggregate (linkagg) ID are configurable as a DHL link.
- A group of VLANs (or pool of common VLANs) in which each VLAN is associated (802.1q tagged) with both link A and link B.
- A VLAN-to-link mapping that specifies which of the common VLANs each DHL link will service. This mapping prevents network loops by designating only one active link for each VLAN, even though both links remain active and are associated with each of the common VLANs.

When one of the two active DHL links fails or is brought down, the VLANs mapped to that link are then forwarded on the remaining active link to maintain connectivity to the core. When the failed link comes back up, DHL waits a configurable amount of time before the link resumes forwarding of its assigned VLAN traffic.

The following diagram shows how DHL works when operating in a normal state (both links up) and when operating in a failed state (one link is down):



Protected VLANs

A protected VLAN is one that is assigned to both links in a DHL session. This means that if the link to which the VLAN is mapped fails, the VLAN is moved to the other active DHL link to maintain connectivity with the core switches.

Any VLAN that is only assigned to one of the DHL links is considered an unprotected VLAN. This type of VLAN is not eligible for DHL support if the link to which the VLAN is assigned fails.

DHL Port Types

DHL is supported on the following port types:

- Physical switch ports.
- Logical link aggregate ports (linkagg ID).
- LPS ports
- NNI ports
- IPM VLAN ports
- DHCP Snooping ports
- IP Source filtering ports.

DHL is not supported on the following port types:

- Any port that is a member of a link aggregate.
- Mobile ports
- 802.1x ports
- GVRP ports.
- UNI ports
- Ports that are enabled for transparent bridging.

Note. No CLI error message is displayed when DHL is configured using a port type that is not supported.

DHL Pre-Emption Timer

The DHL pre-emption timer specifies the amount of time to wait before a failed link that has recovered can resume servicing VLANs that are mapped to that link. This time value is configured on a per-DHL session basis.

MAC Address Flushing

Spanning Tree flushes the MAC address table when a topology change occurs that also changes the forwarding topology. The MAC addresses are then relearned according to the new forwarding topology. This prevents MAC address entries from becoming stale (entries contain old forwarding information).

When a port is configured as a DHL Active-Active link, Spanning Tree is automatically disabled on the port. Since Spanning Tree is not used, a changeover from one DHL link to the other does not trigger a

topology change event and the MAC address table is not automatically flushed. This can create stale MAC address entries that are looking for end devices over the wrong link.

To avoid stale MAC address entries in the forwarding tables of the core switches, some type of communication needs to occur between the edge uplink switch and the core switches. The DHL Active-Active feature provides two methods for clearing stale MAC address entries: MVRP Enhanced Operation or Raw Flooding. Selecting which one of these methods to use is done on a per-DHL session basis.

MVRP Enhanced Operation

The switch uses an enhanced Multiple VLAN Registration Protocol (MVRP) operation to refresh core MAC address tables as follows:

- For each uplink port, the switch will only issue joins for each VLAN that is active on that port. This causes the core switch to only register those VLANs that are active on each link based on the DHL configuration.
- When one of the DHL links fails, the other link will issue joins to establish connectivity for the VLANs that were serviced by the failed link. These new joins will have the “new” flag set, which are forwarded by the core devices and will trigger a flush of the MAC addresses on the core network for the joined VLANs.
- When a failed DHL link recovers, the link will issue new joins to re-establish connectivity for the VLANs the link was servicing before the link went down. These new joins also trigger a flush of the MAC addresses on the core network for the joined VLANs.

The switch interacts normally with the core and other devices for MVRP, treating the DHL VLANs on each uplink port as a fixed registration. This approach requires core devices that support MVRP.

Raw Flooding

When a DHL link fails or recovers and Raw Flooding is enabled for the DHL session, the switch performs the following tasks to trigger MAC movement:

- Identify a list of MAC addresses within the effected VLANs that were learned on non-DHL ports (MAC addresses that were reachable through the effected VLANs).
- Create a tagged packet for each of these addresses. The SA for the packet is one of the MAC addresses from the previously-generated list; the VLAN tag is the resident VLAN for the MAC address; the DA is set for broadcast (all Fs); the body is just filler.
- Transmit the generated packet once for each VLAN-MAC address combination. These packets are sent on the link that takes over for the failed link or on a link that has recovered from a failure.

The MAC movement triggered by the Raw Flooding method should clear any stale MAC entries. However, flooded packets are often assigned a low priority and the switch may filter such packets in a highly utilized network.

DHL Configuration Guidelines

Review the following guidelines before attempting to configure a DHL setup:

- Make sure that DHL linkA *and* linkB are associated with each VLAN that the DHL session will protect. Any VLAN not associated with either link or only associated with one of the links is unprotected.
- DHL linkA *and* linkB should belong to the same default VLAN. In addition, select a default VLAN that is one of the VLANs that the DHL session will protect. For example, if the session is going to protect VLANs 10-20, then assign one of those VLANs as the default VLAN for linkA and linkB.
- Only one DHL session per switch is allowed. Each session can have only two links (linkA and linkB). Specify a physical switch port or a link aggregate (linkagg) ID as a DHL link. The same port or link aggregate is not configurable as both linkA or linkB.
- The administrative state of a DHL session is not configurable until a linkA port and a linkB port are associated with the specified DHL session ID number.
- Spanning Tree is automatically disabled on each link when the DHL session is enabled.
- Do not change the link assignments for the DHL session while the session is enabled.
- Configuring a MAC address flush method (MVRP or Raw Flooding) is recommended if the DHL session ports span across switch modules or the DHL ports are on the same module but the data port is on a different module. Doing so will improve convergence time.
- To improve convergence time for uni-directional traffic, specify Raw Flooding as the MAC flush method for the DHL session.
- Enabling the registrar mode as “forbidden” is recommended before MVRP is enabled on DHL links.

Configuring DHL Active-Active

Configuring a DHL Active-Active setup requires the following tasks.

- 1 Configure a set of VLANs that the two DHL session links will service.

```
-> vlan 100-110
```

- 2 Identify two ports or link aggregates that will serve as the links for the DHL session then assign both links to the same default VLAN. Make sure the default VLAN is one of the VLANs created in Step 1. For example, the following commands assign VLAN 100 as the default VLAN for port 1/10 and linkagg 5:

```
-> vlan 100 port default 1/10  
-> vlan 100 port default 5
```

- 3 Associate (802.1q tag) the ports identified in Step 2 to each one of the VLANs created in Step 1, except for the default VLAN already associated with each port. For example, the following commands associate port 1/10 and linkagg 5 with VLANs 101-110:

```
-> vlan 101-110 802.1q 1/10  
-> vlan 101-110 802.1q 1
```

In the above command example, port 1/10 and linkagg 1 are only tagged with VLANs 101-110 because VLAN 100 is already the default VLAN for both ports.

4 Create a DHL session using the **dhl num** command. For example:

```
-> dhl num 10
```

5 Configure the pre-emption (recovery) timer for the DHL session using the **dhl num pre-emption-time** command. By default, the timer is set to 30 seconds, so it is only necessary to change this parameter if the default value is not sufficient. For example, the following command changes the timer value 500 seconds:

```
-> dhl num 10 pre-emption-time 500
```

6 Configure the MAC address flushing method for the DHL session using the **dhl num mac-flushing** command and specify either the **raw** or **mvrp** parameter option. By default, the MAC flushing method is set to none. For example, the following command selects the MVRP method:

```
-> dhl num 10 mac-flushing mvrp
```

7 Configure two links (linkA and linkB) for the DHL session using the **dhl num linka linkb** command. Specify the ports identified in Step 1 as linkA and linkB. For example:

```
-> dhl num 10 linka linkagg 1 linkb port 1/10
```

8 Select VLANs from the set of VLANs created in Step 2 and map those VLANs to linkB using the **dhl num vlan-map linkb** command. Any VLAN not mapped to linkB is automatically mapped to linkA. By default, all VLANs are mapped to linkA. For example, the following command maps VLANs 11-20 to linkB:

```
-> dhl num 10 vlan-map linkb 11-20
```

9 Administratively enable the DHL session using the **dhl num admin-state** command. For example:

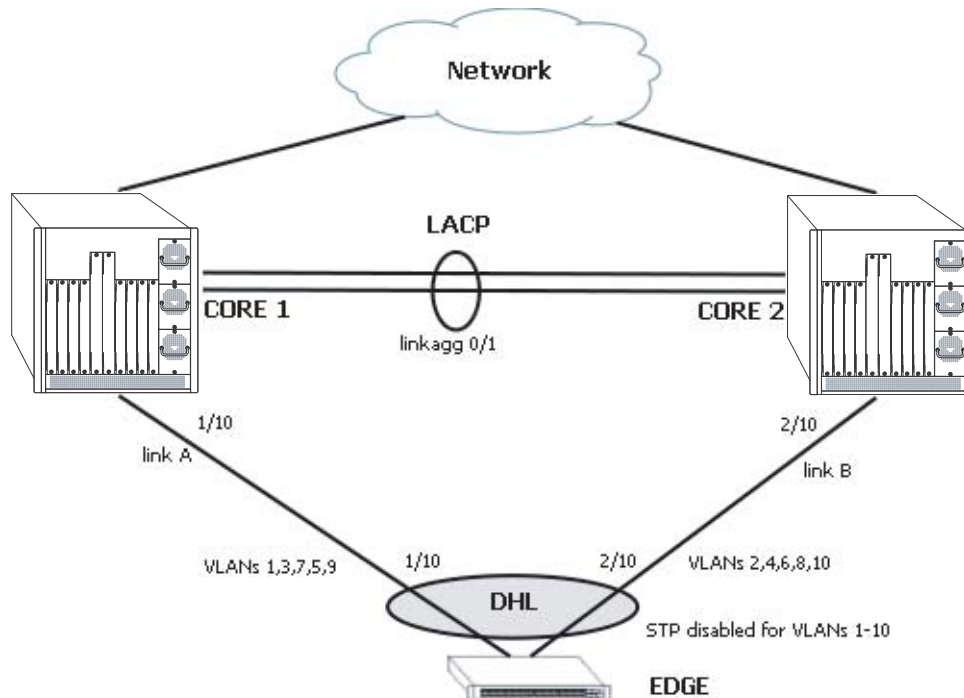
```
-> dhl num 10 admin-state enable
```

See [“Dual-Home Link Active-Active Example”](#) on page 27-10 for a DHL application example.

Dual-Home Link Active-Active Example

The figure below shows two ports (1/10 and 2/10) that serve as link A and link B for a DHL session configured on the Edge switch. Both ports are associated with VLANs 1-10, where VLAN 1 is the default VLAN for both ports. The odd numbered VLANs (1, 3, 5, 7, 9) are mapped to link A and the even numbered VLANs (2, 4, 6, 8, 10) are mapped to link B. Spanning Tree is disabled on both ports.

Both DHL links are active and provide connectivity to the Core switches for the VLANs to which each link is mapped. If one link fails or is brought down, the VLANs mapped to the failed link are switched over to the remaining active link to maintain connectivity for those VLANs. For example, if link A goes down, VLANs 1, 3, 5, 7, and 9 are switched over and carried on link B.



Dual-Home Link Active-Active Example

Follow the steps below to configure this example DHL configuration.

Edge Switch:

- 1 Create VLANs 2-10.

```
-> vlan 2-10
```

- 2 Configure 802.1q tagging on VLANs 2-10 for port 1/10. Because VLAN 1 is the default VLAN for port 1/10, there is no need to tag VLAN 1.

```
-> vlan 2-10 802.1q 1/10
```

- 3 Configure 802.1q tagging on the VLANs 2-10 for port 2/10. Because VLAN 1 is the default VLAN for port 2/10, there is no need to tag VLAN 1.

```
-> vlan 2-10 802.1q 2/10
```

- 4 Configure a session ID and an optional name for the DHL session.

```
-> dhl num 1 name dhl_session1
```

- 5 Configure port 1/10 and port 2/10 as the dual-home links (linkA, linkB) for the DHL session.

```
-> dhl num 1 linkA port 1/10 linkB port 2/10
```

- 6 Map VLANs 2, 4, 6, 8, and 10 to DHL linkB.

```
-> dhl num 1 vlan-map linkb 2
-> dhl num 1 vlan-map linkb 4
-> dhl num 1 vlan-map linkb 6
-> dhl num 1 vlan-map linkb 8
-> dhl num 1 vlan-map linkb 10
```

- 7 Specify Raw Flooding as the MAC flushing technique to use for this DHL session.

```
-> dhl num 1 mac-flushing raw
```

- 8 Enable the administrative state of the DHL session using the following command:

```
-> dhl num 1 admin-state enable
```

Core Switches:

- 1 Create VLANs 2-10.

```
-> vlan 2-10
```

- 2 Configure 802.1q tagging on VLANs 2-10 for port 1/10 on the Core 1 switch. VLAN 1 is the default VLAN for port 1/10, so there is no need to tag VLAN 1.

```
-> vlan 2-10 802.1q 1/10
```

- 3 Configure 802.1q tagging on VLANs 2-10 for port 2/10 on the Core 2 switch. VLAN 1 is the default VLAN for port 2/10, so there is no need to tag VLAN 1.

```
-> vlan 2-10 802.1q 2/10
```

- 4 Configure 802.1q tagging on VLANs 2-10 for LACP 1 on both of the Core switches. VLAN 1 is the default VLAN for LACP 1 on both Core switches, so there is no need to tag VLAN 1.

```
-> vlan 2-10 802.1q 1
```

CLI Command Sequence Example

The following is an example of what the example DHL configuration commands look like entered sequentially on the command line:

Edge Switch:

```
-> vlan 2-10
-> vlan 2-10 802.1q 1/10
-> vlan 2-10 802.1q 2/10
-> dhl num 1 name dhl_session1
-> dhl num 1 linkA port 1/10 linkB port 2/10
-> dhl num 1 vlan-map linkb 2
-> dhl num 1 vlan-map linkb 4
-> dhl num 1 vlan-map linkb 6
-> dhl num 1 vlan-map linkb 8
-> dhl num 1 vlan-map linkb 10
-> dhl num 1 mac-flushing raw
-> dhl num 1 admin-state enable
```

Core 1 Switch:

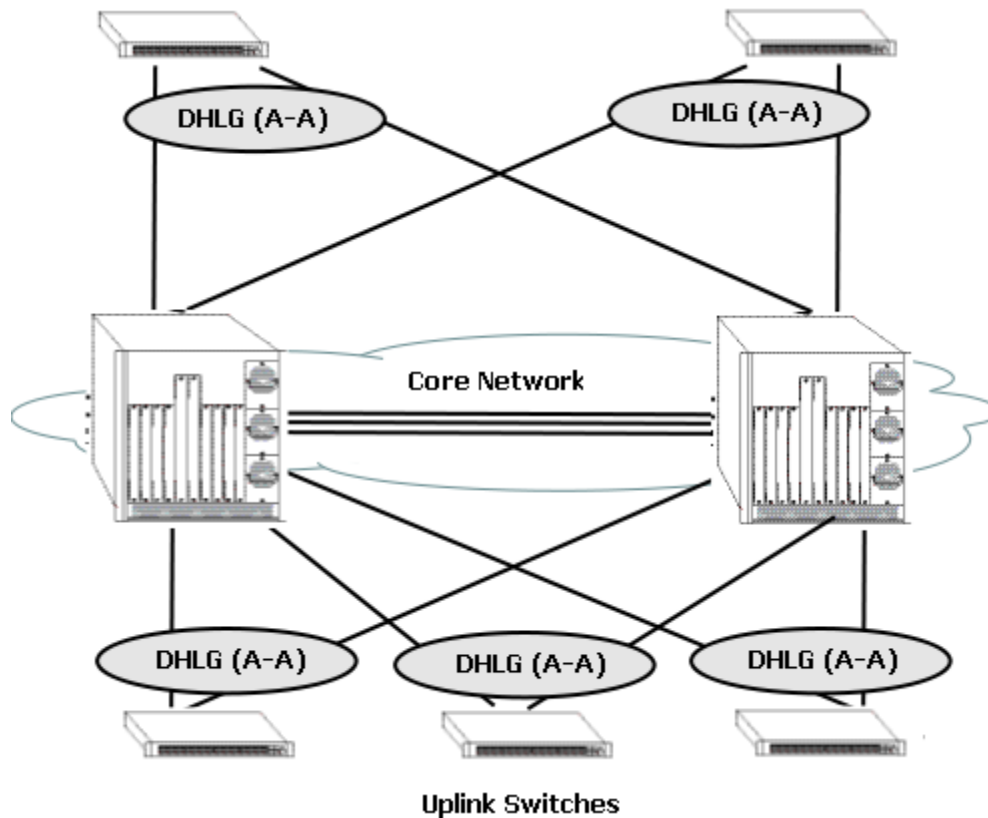
```
-> vlan 2-10
-> vlan 2-10 802.1q 1/10
-> vlan 2-10 802.1q 1
```

Core 2 Switch:

```
-> vlan 2-10
-> vlan 2-10 802.1q 2/10
-> vlan 2-10 802.1q 1
```

Recommended DHL Active-Active Topology

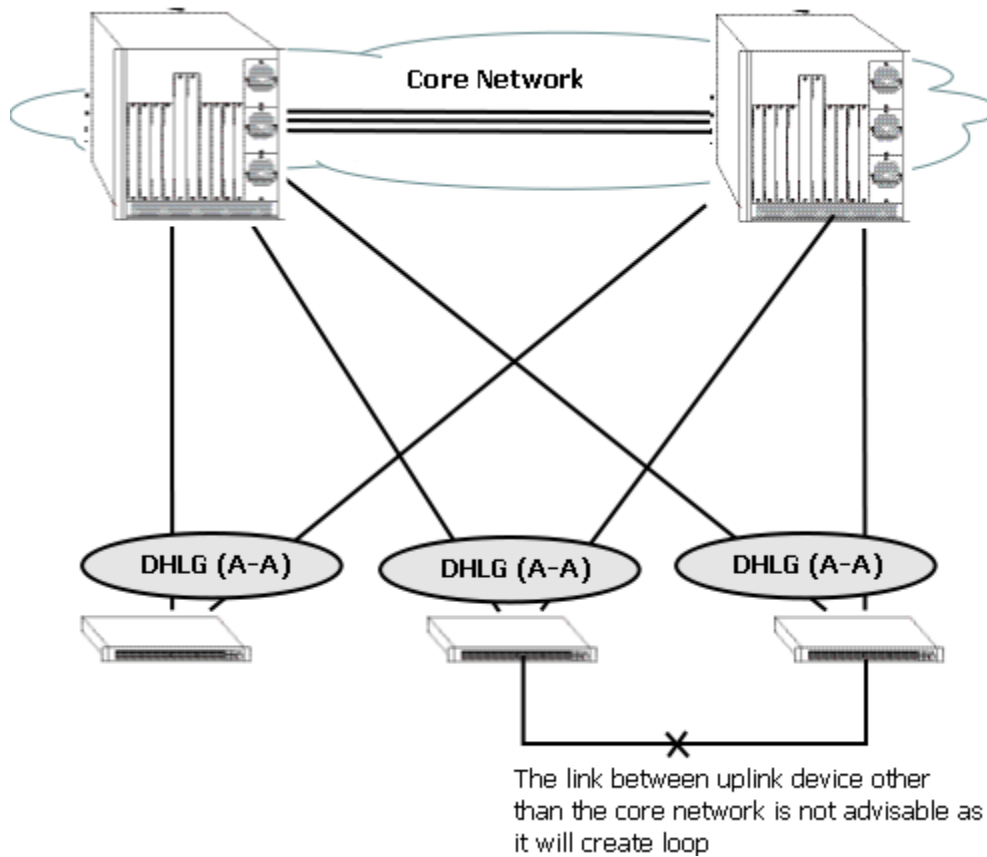
The following is an example of a recommended topology for Dual-Home Link Active-Active.



In the above topology, all uplinked switches are connected to the core network through redundant links, and the links are configured to use DHL Active-Active. Spanning Tree is disabled on all the DHL enabled ports of the uplinked devices.

Unsupported DHL Active-Active Topology (Network Loops)

The following is an example of an unsupported topology for Dual-Homed Link Active-Active.



In the above topology, the link between the uplink device other than core network is not recommended as it will create a loop in the network. This topology violates the principle that uplink switches can only be connected to the network cloud through the core network.

Displaying the Dual-Home Link Configuration

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. |
| show dhl | Displays the global status of the dhl configuration. |
| show dhl num | Displays information about a specific DHL session. |
| show dhl num link | Displays information about a specific DHL link, for example linkA or linkB and the VLAN details of the specified link. |

Note. See the “Link Aggregation Commands” chapter in the *OmniSwitch AOS Release 6 CLI Reference Guide* for complete documentation of **show** commands for link aggregation.

28 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,
- 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 29, “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 (for example, `ip default-ttl`). CLI commands are used in the configuration examples; for more details about the syntax of commands, see the *OmniSwitch AOS Release 6 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 28-7](#))
 - Creating a Static Route (see [page 28-10](#))
 - Creating a Default Route (see [page 28-12](#))
 - Configuring Address Resolution Protocol (ARP) (see [page 28-13](#))
- IP Configuration
 - Configuring a DHCP Client Interface (see [page 28-18](#))
 - Configuring the Router Primary Address (see [page 28-18](#))
 - Configuring the Router ID (see [page 28-18](#))
 - Configuring the Time-to-Live (TTL) Value (see [page 28-19](#))
 - Configuring Route Map Redistribution (see [page 28-19](#))
 - IP-Directed Broadcasts (see [page 28-26](#))
 - Protecting the Switch from Denial of Service (DoS) attacks (see [page 28-27](#))

- Managing IP
 - Internet Control Message Protocol (ICMP) (see [page 28-34](#))
 - Using the Ping Command (see [page 28-37](#))
 - Tracing an IP Route (see [page 28-38](#))
 - Displaying TCP Information (see [page 28-38](#))
 - Displaying User Datagram Protocol (UDP) Information (see [page 28-38](#))
 - Two-Way Active Measurement Protocol (TWAMP) (see [page 28-39](#))
- Network Address Translation
 - Configuring NAT (see [page 28-42](#))

IP Specifications

Note. 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 RFC 5357–Two-Way Active Measurement 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 6350, 6450 |
| Maximum VLANs per switch | 4094 |
| Maximum IP interfaces per switch | 128 32 (OS6350) |
| Maximum IP interfaces per VLAN | 32 8 (OS6350) |
| Maximum ARP entries per switch | 512 256 (OS6350) |
| Maximum ARP filters per switch | 200 |
| Maximum length of IP interface name | 32 |
| Maximum IP static routes per switch | 256 64 (OS6350) |
| Maximum IP host routes per switch | 256 |
| Maximum number of ECMP gateways (per static route) | 4 |
| Maximum TWAMP control sessions per switch | 32 |

| | |
|--|-----|
| Maximum TWAMP test sessions per switch | 128 |
|--|-----|

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 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, 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 (for example, VLAN 1) by using the **vlan** command. For example:

```
-> vlan 1 name "VLAN 1"
```

- 2 Create VLAN 2 with a description (for example, 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 in the following sections.

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 can be damaged in transit, thrown away by a busy switch, or never reach its destination. The resolution for 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 32, “Configuring DHCP.”](#)

Application-Layer Protocols

Application-layer protocols are used for switch configuration and management:

- Bootstrap Protocol (BOOTP)/Dynamic Host Configuration Protocol (DHCP)—can 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 AOS Release 6 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 can 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 28-13](#).
- 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 28-34](#).
- Router Discovery Protocol (RDP)—Used to advertise and discover routers on the LAN. For more information, see [Chapter 31, “Configuring RDP.”](#)
- Multicast Services—Includes IP multicast switching (IPMS). For more information, see [Chapter 41, “Configuring IP Multicast Switching.”](#)

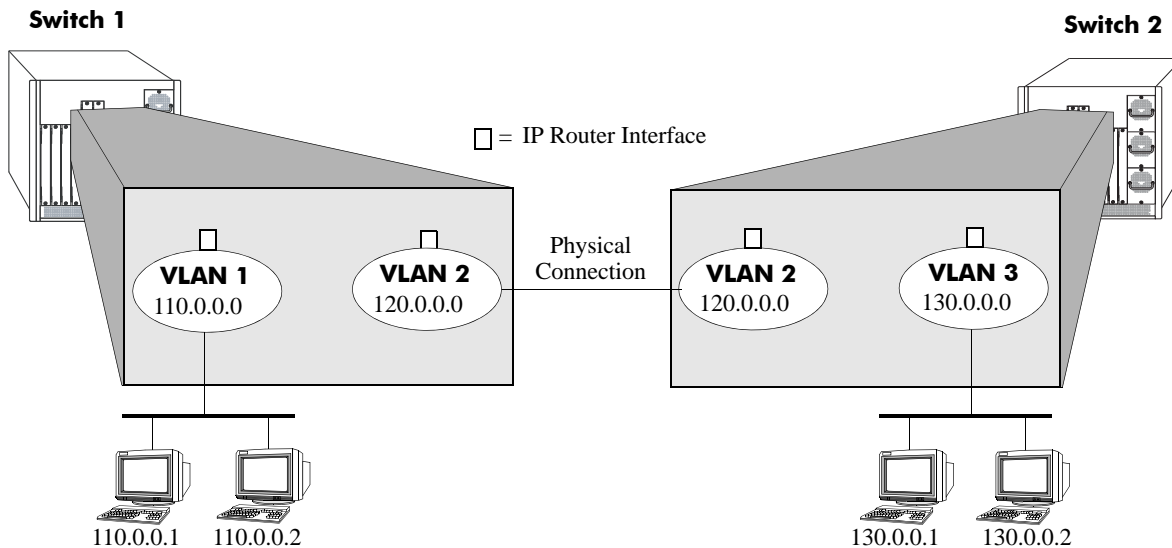
IP Forwarding

Network device traffic is bridged or 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, Layer 3 routing is used to transmit traffic between the VLANs. Bridging decides on where to forward packets based on the packet destination MAC address; routing decides on where to forward packets based on the packet IP network address (for example, IP - 21.0.0.10).

Alcatel 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 32 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 following illustration, 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. Hence, 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, 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 (for example, 193.204.173.21).

Note. The 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 (for example, 255.255.0.0) or with a slash (/) after the IP address followed by the number of bits to specify the mask length (for example, 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 28-14](#).
- 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 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 remove the IP interface and then create it again with the new values. The changes specified overwrite the 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. When the IP address for the interface is changed, the subnet mask reverts to the default mask value if it was previously set to a non-default value and no mask value is specified when the IP address is changed.

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 previous 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, 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 AOS Release 6 CLI Reference Guide*.

Removing an IP Router Interface

To remove an IP router interface, use the **no** form of the **ip interface** command.

Note. It is only necessary to specify the name of the IP interface, along with the **no** form of the command.

For 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 AOS Release 6 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 remains 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. 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. 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
```

Gateway and destination IP address must be on different subnets. 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, 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. You can also specify the length of the mask in bits.

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; hence, delete them from the table. Use the **no ip static route** command to delete a static route. 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
```

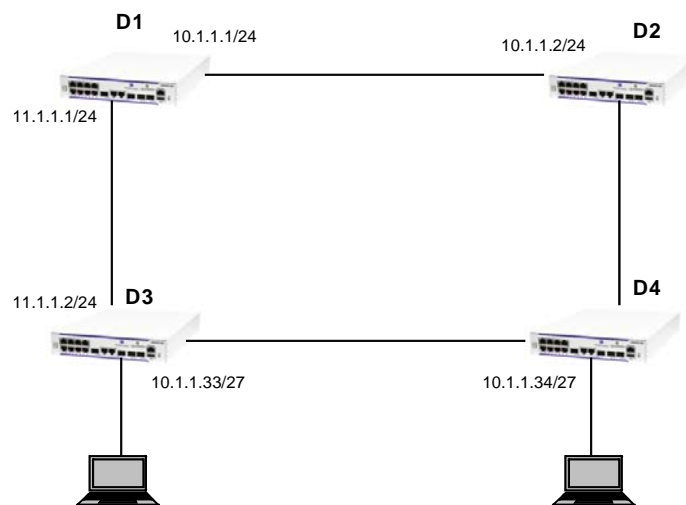
The IP Forwarding table includes routes learned through RIP as well as any static routes that are configured. Use the **show ip route** command to display the IP Forwarding table.

Note. A static route is not active unless the gateway it is using is active.

Ability to add static routes in a directly connected network

Multiple static routes can be added to a subnet of directly connected network.

For example, in the following scenario, D1 has a local route 10.1.1.1/24. Another static route 10.1.1.33/27 can be added with a gateway address 11.1.1.2/24. Switch supports two routes, a local route to 10.1.1.0/24 with 10.1.1.1/24 as the gateway, and a static route to 10.1.1.32/27 with 11.1.1.2/24 as the gateway. Packets destined to 10.1.1.32/27 network are routed through 11.1.1.2/24 and packets to other IP address or subnets in 10.1.1.0/24 are routed through 10.1.1.1/24.



Use **show ip route** command to view the configured static routes.

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. 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. You can specify the length of the mask in bits also.

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. 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 can also specify:

- **Alias.** Use the **alias** keyword to configure the switch to 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. 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 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 through the router. ARP packets are still bridged across multiple ports.

Note. 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. 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.

Dynamic Proxy ARP - Mac Forced Forwarding

Dynamic Proxy ARP - MAC Forced Forwarding is used to forward all traffic from L2 clients to a head-end router. This head-end router filters and forwards the traffic from the local network or back to other clients in the same VLAN/IP subnet. In order to accomplish this, Dynamic Proxy ARP combines the functionality of other switch features to dynamically learn router addresses and act as a proxy for that router. Dynamic Proxy ARP - MAC Forced Forwarding uses the following features:

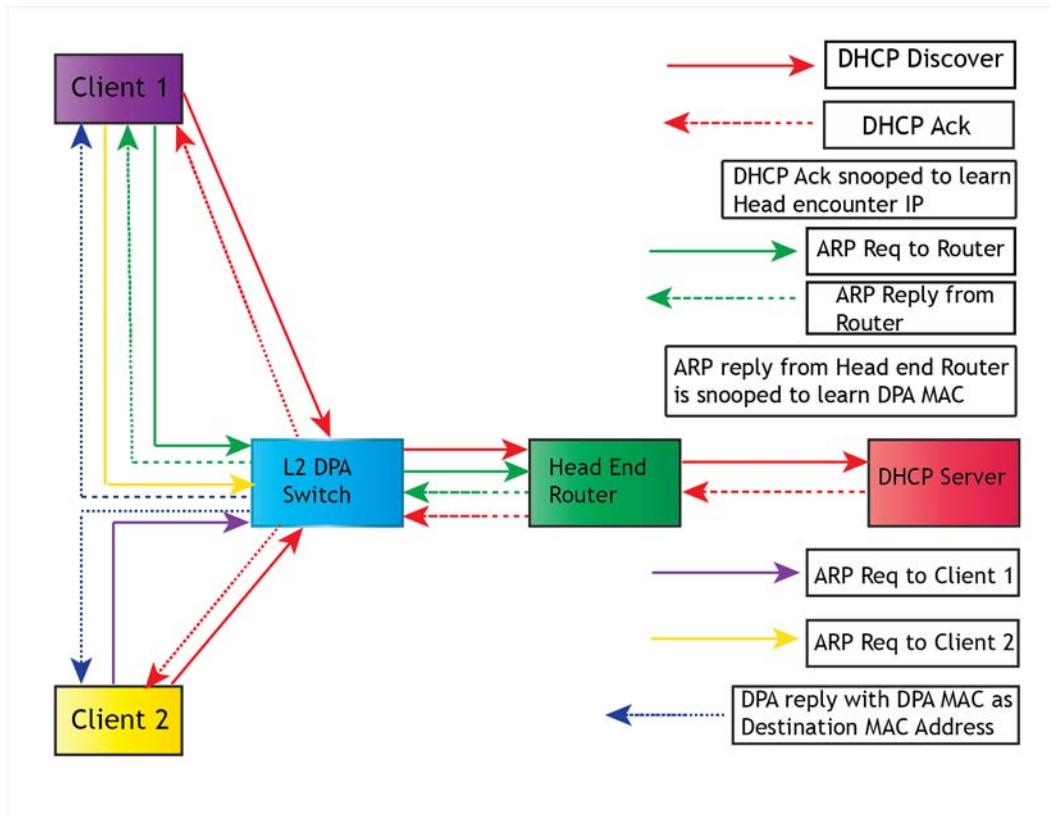
Port Mapping - Port Mapping forwards traffic from user-ports only to network-ports, preventing communication between L2 clients in the same VLAN. Port mapping prevents direct communication between clients in the same VLAN forcing all traffic to be forwarded to the head end router.

Proxy ARP - All ARP requests received on port mapping user-ports are answered with the MAC address of the head end router. Dynamic Proxy ARP dynamically learns the IP and MAC address of a head end router and responds with that router MAC address instead of flooding the ARP request.

DHCP Snooping - Snoops the DHCP packets between the server and clients. DHCP snooping is used to dynamically learn the IP address of the head end router.

MAC Forced Forwarding Steps:

1. Clients are connected to the user-ports of a port mapping session.
2. Head end router is connected to the network-port of the same port mapping session.
3. DHCP snooping is enabled and uses the DHCP DISCOVER and DHCP ACK packets to learn the head end router IP.
4. The ARP Request and Reply is snooped by the switch to learn the head end router MAC address.
5. The ARP Requests from clients on the user-ports are intercepted by the switch and the switch replies with the head end router MAC address.
6. All traffic from the clients is now forwarded to the head end router to be filtered.



Dynamic Proxy ARP

Use the **port mapping user-port network-port** and **ip helper dhcp-snooping vlan** commands as follows to enable Dynamic Proxy ARP - MAC Forced Forwarding. For example:

```
-> port mapping 1 user-port 1/1-2 network-ports 1/3
-> port mapping 1 dynamic-proxy-arp enable
-> ip helper dhcp-snooping vlan 1
```

The example above considers that all devices are in VLAN 1, Clients 1 and 2 are connected to ports 1/1 and 1/2, and the head end router is connected to port 1/3.

ARP Filtering

ARP filtering is used to determine whether the switch responds to ARP requests that contain a specific IP address. ARP filtering feature is 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 **arp filter** command to specify the following parameter values required to create an ARP filter:

- An IP address (for example, 193.204.173.21) used to determine whether an ARP packet is filtered.
- An IP mask (for example 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, that blocks 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 on all supported ARP filter commands, see the *OmniSwitch AOS Release 6 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, enter:

```
-> ip interface dhcp-client vlan 100
```

Refer to the [“Configuring the DHCP Client Interface” on page 32-17](#) 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 can 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. The route-map 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 28-20](#).
- 2 Configure redistribution to apply a route map, as described in [“Configuring Route Map Redistribution” on page 28-24](#).

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 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. The **Set** 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 AOS Release 6 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 28-24 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 `redstripv4`:

```
-> no ip route-map redstripv4
```

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 `redstripv4` route map:

```
-> no ip route-map redstripv4 sequence-number 10
```

Note that in this example, the `redstripv4` 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 `redstripv4` sequence 10:

```
-> no ip route-map redstripv4 sequence-number 10 match tag 8
```

Configuring Route Map Sequences

A route map can 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 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 can contain multiple match statements. If these statements are of the same kind (for example, match tag 5, match tag 8, and so on) then a logical OR is implied between each like statement. If the match statements specify different types of matches (for example match tag 5, match ip4 interface to-finance, and so on), then a logical AND is implied between each statement. For example, the following route map sequence redistributes 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 redistributes 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 mentioned 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 AOS Release 6 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 performs 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 can also specify the modification of route information before the route is redistributed. See [“Using Route Maps” on page 28-20](#) 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 (for routes 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 zeros 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. Ideally, directed broadcasts must 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.

Controlled Directed Broadcasts

The Control Directed Broadcast can be configured to support subnet-directed broadcast, which is allowed only for a given/trusted set of source IP address, destination IP address, and VLANs. Use the **controlled** keyword in the **ip directed-broadcast** command to broadcast only the IP packets received from the user defined source.

```
-> ip directed-broadcast controlled
```

Specify the source IP address, destination IP address, and VLAN information to broadcast the packets in controlled manner by using **ip directed-broadcast allow** command. The specified information is considered as the trusted information to broadcast the packets received only from the defined source, and the remaining broadcast packets are dropped.

For example:

```
-> ip directed-broadcast allow source-ip 30.0.0.10/24 destination-ip 10.0.0.255/  
24
```

```
-> ip directed-broadcast allow source-ip 30.0.0.10/24 vlan 10
```

```
-> ip directed-broadcast allow source-ip 30.0.0.10/24 destination-ip 10.0.0.255/  
24 vlan 10
```

Use the **show ip config** command to display the IP directed broadcast state. This command also displays the source IP address, destination IP address, and VLAN information of the control directed broadcast.

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 is 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 can 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 as follows:

| | |
|---------------------------|--|
| Invalid Source IP address | <ul style="list-style-type: none">• 0.x.x.x.• 255.255.255.255.• subnet broadcast, 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 for example). • 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.

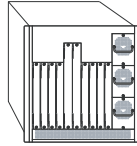
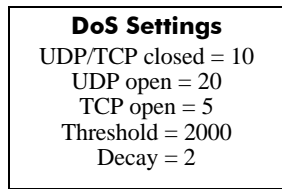
Note. In both the conditions described here in “Multicast IP and MAC Address Mismatch”, packets are dropped and SNMP traps are generated.

- 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 (loop-back 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 is 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:

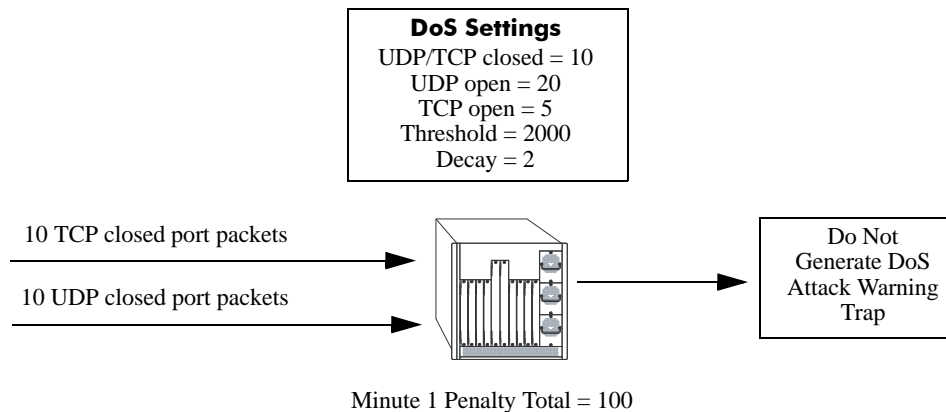


Penalty Total = 0

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:

$$(10 \text{ TCP} \times 10 \text{ penalty}) + (10 \text{ UDP} \times 10 \text{ penalty}) = 200$$

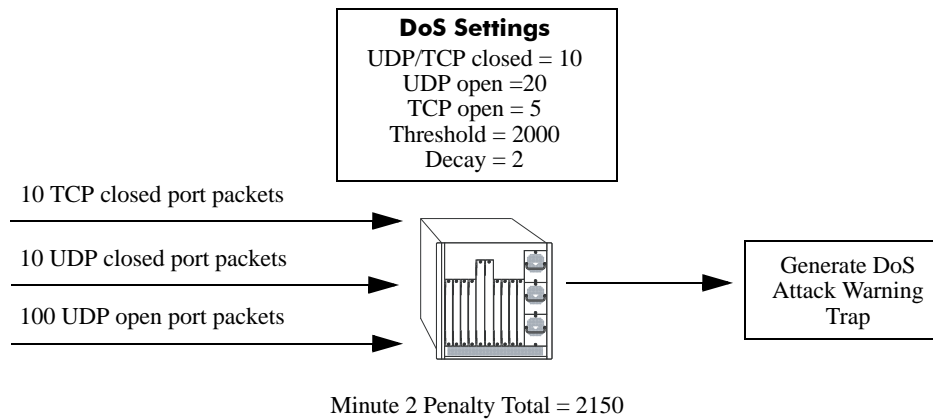
This value would be divided by 2 (due to the decay) and decreased to 100. The switch would not record a port scan:



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 function usage 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 is in progress when the switch 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 does 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 AOS Release 6 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, and so on, 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 |

Extend IPv4 Interfaces and Static Routes Support on OmniSwitch

Use **ip tables extend** command to obtain the space to extend the number of IPv4 interfaces and IPv4 static routes supported on the switch by reducing the number of IPv6 neighbor entries to 76.

After using this command, save the configurations using the **write memory** command and reload the switch to reflect the revised space allocation for interface and static routes.

```
-> ip tables extend
```

Use **ip tables default** to enable default allocation of IPv4 interfaces and IPv4 static routes. By default, 8 interfaces, 8 static routes, and 96 IPv6 neighbor entries are supported.

```
-> ip tables default
```

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 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 TTL field reaches zero. The TTL field prevents packets from continuously circulating the internetwork if the internetwork contains a routing loop. Once a packet 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 provided 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 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 34, “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 IP address or host name. The switch pings 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 (maximum of 19 characters) is required. You can also specify the following parameters:

- **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 uses to poll the host.
- **time-out** -- Use the time-out keyword to set the number of seconds the program must wait for a response before timing out.
- **source-ip** -- Use the source-ip keyword to set the source IP address for receiving the ping packets. The source IP can also be a Loopback address
- **tos** -- Use the tos keyword to set the type of service value for the ping packet being transmitted.
- **dont-fragment** -- Use the dont-fragment keyword to set the DF bit in the transmitted ping packet

- **pattern** - Use the pattern keyword to set the data pattern string
-> ping 10.0.0.1 pattern AB1234
- **sweep-range** - Use the sweep-range keyword to set the sweep range size. The sweep range requires 3 parameters. start_size specifies the size in bytes of the first packet to be sent, diff_size specifies the increment factor of size for the next packet, and end_size specifies the maximum size of the packet. Here, if sweep-range is used in the ping command, then the count and size parameters become redundant. So if sweep-range is used, then count and size parameters are not configurable. Also the values for minsize (greater than 4 bytes) and maxSize (greater than minimum size) are validated.
-> ping 10.0.0.1 sweep-range 10 110 20

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 only apply to the current ping. The next time you use the **ping** command, the default values are used unless you enter different values again.

Tracing an IP Route

The **tracroute** 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, 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.

Displaying Probe Information

The probe information The destination port number to be used in the probing packets. The value must be greater than 1024. The probe value is incremented by one in each probe. The default port number used is 33334 (32768+666).

Two-Way Active Measurement Protocol (TWAMP)

The Two-Way Active Measurement Protocol (TWAMP) is an open protocol for measurement of two-way metrics between any two network devices which supports the TWAMP protocol. TWAMP provides a standard technique to measure network performance metrics. Unlike ICMP Ping, TWAMP also measures round trip delay/Jitter apart from the RTT.

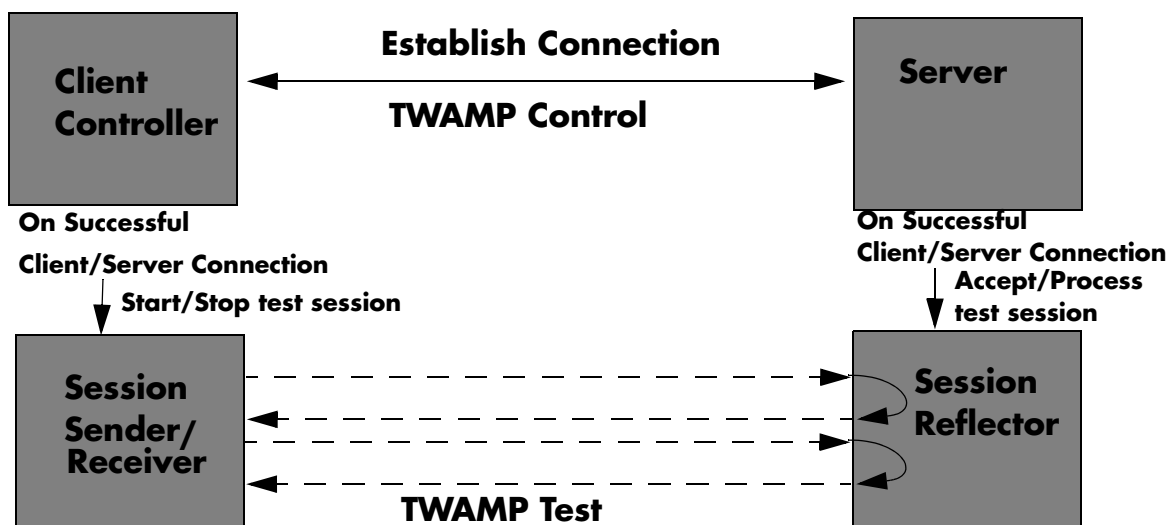
TWAMP does not use clock synchronization between the two devices instead employs time stamps applied at the echo destination (reflector) to enable greater accuracy.

TWAMP works on the client/server model. The client is the sender and receiver of the test sessions, the switch which is configured for TWAMP acts as server and reflector.

TWAMP on OmniSwitch

The TWAMP protocol operates based on four logical entities Server, Control Client, Sender and Reflector. TWAMP protocol uses TWAMP Control and TWAMP Test:

- TWAMP Control, establishes a session between the client and the server.
- TWAMP Test, manages the test session between the sender and receiver.



The switch can be configured to manage the TWAMP sessions. The current TWAMP implementation supports only:

- Server and Reflector configuration
- Unauthenticated mode
- IPv4 address
- OmniSwitch 6350 and OmniSwitch 6450

In the TWAMP Server/Reflector Mode Operation, on configuring the TWAMP server on the switch, the switch will:

- Respond to TCP open messages from various clients and establish TWAMP control connection.
- Accept TWAMP test session requests from clients and manage the sessions.
- Respond to the TWAMP test packets with TWAMP reflector test packets.

Note. Maximum 32 control sessions are allowed per switch. A control session can have 128 test sessions. Maximum 128 test sessions are allowed per switch.

TWAMP Operation

TWAMP Control connection must be established between the Control Client and Server before initiating any test sessions. Control Connection establishment involves the following steps:

- 1** The Control-Client initiates a TCP connection on TWAMP's well-known port 862 or the user configured port, and the server responds with greeting message, indicating the security/integrity mode(s) it is willing to support. Currently, only unauthenticated mode is supported.
- 2** The Control-Client responds with the chosen mode of communication and information supporting integrity protection and encryption, if the mode requires them.
- 3** The Server responds to accept the mode and give its start time. This completes the control-connection setup.
- 4** The Control-Client requests a test session with a unique TWAMP-Control message. The Server responds with its acceptance and supporting information. More than one test session may be requested with additional messages.
- 5** The Control-Client initiates all requested testing with a Start-Session message, and the Server acknowledges.
- 6** The Session-Sender and the Session-Reflector exchange test packets according to the TWAMP-Test protocol for each active session.
- 7** When appropriate, the Control-Client sends a message to stop all test sessions.

TWAMP Metrics Measurement

When the test session is acknowledged by the server, the session-sender starts sending test frames to the reflector with its own timestamp, sequence number and so on. The reflector performs the following steps:

- 1** Timestamp the received packet on ingress as close to hardware as possible.
- 2** Copy the packet sequence number and the timestamp of sender.
- 3** Extract the TTL from IP Header.
- 4** Encode a reflector packet with the Rx, Tx timestamp of the reflector, sequence number and the sender fields timestamp, sequence number and TTL.
- 5** Transmit the reflector packet for every test packet received.
- 6** The session sender will receive the reflector frames and calculate the RTT, jitter, and delay metrics for the session.

Configuring TWAMP Server on the Switch

Configuring the TWAMP server on the switch involves configuring the TCP port to be used by the TWAMP server, configuring the client which shall be allowed to establish connection with the TWAMP server, and configuring the inactivity timer for the control session.

Use the `twamp server` command to configure the TWAMP server on the switch. For example, to configure the TCP port 30333 with allowed client access to 172.16.1.1/16, and control session inactivity timer of 20 seconds, enter:

```
-> twamp server port 30333 inactivity-timeout 20 allowed-client 172.16.1.1/16
```

Note.

- When the TWAMP server is configured on the switch, the loopback0 IP address will be taken as the IP address of the server.
- Only one TWAMP server can be configured on a switch at a given point of time.
- When TWAMP server port is reconfigured, a system reload is needed for the reconfigured port to come into effect.

For more information on the CLI, refer the `twamp server` command in the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Removing the TWAMP Server Configuration from the Switch

To remove the TWAMP server configuration from the switch, use the `no` form of `twamp server` command. For example:

```
-> no twamp server
```

Viewing the TWAMP Server Information and Established Client Connections

The TWAMP server information displays the configuration details of the TWAMP server on the switch. To view the TWAMP server configuration details on the switch, use the `show twamp server info` command. For example:

```
-> show twamp server info

TWAMP Server
Port: 30333
Inactivity timeout: 20 mins
Allowed-Client: 172.16.1.1/16
```

The TWAMP client connection information displays the client connection details such as the client IP address, connection status, time when run, number of packets sent, number of packets received, and the session identifier. The statistics details for the established connections is updated every two minutes. The statistics for maximum 128 test sessions after the timeout value from the client side is updated with the connection status as “ENDED”. To view the TWAMP client connection details, use the `show twamp server connections` command. For example, to view the client connection details for the client with IP 200.200.1.1, enter:

```
-> show twamp server connections client 200.200.1.1
```

| Client IP | Conn Status | Time of Last Run | Pkts Sent | Pkts Received | Session Identifier |
|-------------|-------------|--------------------------|-----------|---------------|--------------------|
| 200.200.1.1 | SETUP_DONE | THU OCT 08 2015 19:39:13 | 10 | 10 | 2eb0b7b6a5c405df |
| 200.200.1.1 | SETUP_DONE | THU OCT 08 2015 19:39:13 | 10 | 10 | 2eb0b7b6fe7fb742 |

Network Address Translation

Network Address Translation (NAT) is a feature that allows an organization's IP network to appear from the outside to use different IP address space than what it is actually using. Thus, NAT allows an organization using private addresses (local addresses, and therefore not accessible through the Internet routing tables), to connect to the Internet by translating those addresses into globally routable address space (public addresses), which are accessible from Internet. NAT also allows organizations to launch readdressing strategies where the changes in the local IP networks are minimum.

NAT is used for rewriting a source or destination IP address to another address. A single address may be rewritten, or an entire subnet or list of IP addresses may be rewritten to a group of addresses.

Following are the functionality provided by the feature:

- Static NAT - the local address is always mapped to the same global address. With static NAT, we can translate between local networks and global networks of the same size (contain the same number of IP addresses).
- Dynamic NAT - establishes a mapping of local addresses to a pool of global addresses. This means that the mapping between local addresses and global addresses will not always be the same. This also allows for a mapped pool of local addresses that is larger than the global addresses.
- NAT (Address Port Translation) - the mapping between local addresses and an unique global address. In this case, a translation of the transport protocols ports (UDP, TCP) is carried out.

Note.

- AOS supports only Many to one Dynamic NAT. Telnet, Ping, SSH, SNMP packets are not considered for NAT packets.
 - NAT is supported only in standalone switches.
 - When NAT is configured, traceroute does not work with public network.
 - For the NAT to work, the IP interface of the switch must be same as the re-write IP.
-

Configuring NAT

To configure NAT, QoS policy rule must be mapped with a policy condition and action using the following commands.

- Enable NAT policy condition for a source or destination IP/network using the **policy condition** command with source or destination IP address which needs to be natted.
- Enable policy action, that is, configure the source or destination IP which needs to be rewritten to global IP, use the **policy action rewrite** command.
- Configure a policy rule to map the NAT condition with the action using the **policy rule** command. For example,

Note. Refer to the “QoS Policy Commands” chapter in the *OmniSwitch AOS Release 6 CLI Reference Guide* for more information on QoS policy rule, policy condition, and policy action commands.

Example 1: Configuring Static NAT through policy condition

- 1 Create a policy rule (trans_rule1) on the switch.
- 2 Enable NAT policy condition (cond1) for the source IP (128.110.124.120) which needs to be rewritten.
- 3 Enable policy action (action1) for the source IP which needs to be rewritten to global IP (155.100.39.163).
- 4 Map the NAT condition with the action to the policy rule that will rewrite the source address.

```
-> policy rule trans_rule1
-> policy condition cond1 source ip 128.110.124.120
-> policy action action1 source rewrite ip 155.100.39.163
-> policy rule trans_rule1 condition cond1 action action1
```

When the traffic arrives on the switch with a source address of 128.110.124.120, the switch will rewrite the source address as 155.100.39.163.

Example 2: Configuring Many-to-one NAT

For this type of translation, a single policy rule is required. A network address is given in the condition, and a single address is given in the action.

- 1 Create a policy rule (nat) on the switch.
- 2 Enable NAT policy condition (internal) for the source IP (IP 10.0.0.0 and mask 255.0.0.0) which needs to be rewritten. Note that the mask must be specified for the source address.
- 3 Enable policy action (external) for the source IP which needs to be rewritten to global IP (143.209.92.42).
- 4 Map the NAT condition with the action to the policy rule that will rewrite the source address.

```
-> policy rule trans_rule1
-> policy condition internal source ip 10.0.0.0 mask 255.0.0.0
-> policy action external source rewrite ip 143.209.92.42
-> policy rule nat condition internal action external
```

The policy **nat** will rewrite the source address for any traffic from the 10.0.0.0 network to the Internet friendly address, 143.209.92.42. Traffic destined for the 10.0.0.0 network will be rewritten to the original IP addresses based on the dynamic TCP/UDP port assignment.

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 route-pref | 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 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. |
| show twamp server info | Displays the configuration details of the TWAMP server on the switch. |
| show twamp server connections | Displays the TWAMP Client connections established with the TWAMP server on the switch at a given point of time. The TWAMP connections can also be viewed for a specific client. |
| show qos nat flows | Displays the flow inbound and outbound traffic details. |

For more information about the displays that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

29 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 AOS Release 6 CLI Reference Guide*.

This chapter provides an overview of IPv6 and includes information about the following procedures:

- Configuring an IPv6 interface (see [page 29-9](#))
- Assigning IPv6 Addresses (see [page 29-11](#))
- Creating a Static Route (see [page 29-12](#))
- Configuring the Route Preference of a Router (see [page 29-13](#))
- Configuring Route Map Redistribution (see [page 29-13](#))
- Configuring Router Advertisement (RA) Filtering (see [page 29-20](#))

IPv6 Specifications

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

| | |
|--|---|
| RFCs Supported | 1981– <i>Path MTU discovery</i> 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> 3595– <i>Textual Conventions for IPv6 Flow Label</i> 4007– <i>IPv6 Scoped Address Architecture</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> 4861– <i>Neighbor discovery for IPv6</i> 4862– <i>IPv6 stateless address auto-configuration</i> |
| Platforms Supported | OmniSwitch 6350, 6450 (IPv6 RIP not supported on 6350) |
| Maximum IPv6 interfaces | 16 4 (OS6350) |
| Maximum IPv6 interfaces per VLAN | 1 |
| Maximum IPv6 global unicast addressess | 16 4 (OS6350) |
| Maximum IPv6 global unicast addresses per IPv6 interface | 10 4 (OS6350) |
| Maximum IPv6 static routes per switch | 128 4 (OS6350) |
| Maximum IPv6 host routes per switch | 128 |
| Maximum IPv6 neighbors (ND) | 128 96 (OS6350) |
| 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 |

Quick Steps for Configuring IPv6 Routing

The following tutorial assumes that VLAN 200 and VLAN 300 already exist in the switch configuration. For information about how to configure VLANs, see [Chapter 4, “Configuring VLANs.”](#)

- 1 Configure an IPv6 interface for VLAN 200 by using the [ipv6 interface](#) command. For example:

```
-> ipv6 interface v6if-v200 vlan 200
```

Note that when the IPv6 interface is configured, the switch automatically generates a link-local address for the interface. This allows for communication with other interfaces and/or devices on the same link, but does not provide routing between interfaces.

- 2 Assign a unicast address to the *v6if-v200* interface by using the [ipv6 address](#) command. For example:

```
-> ipv6 address 4100:1::/64 eui-64 v6if-v200
```

- 3 Configure an IPv6 interface for VLAN 300 by using the [ipv6 interface](#) command. For example:

```
-> ipv6 interface v6if-v300 vlan 300
```

- 4 Assign a unicast address to the *v6if-v300* interface by using the [ipv6 address](#) command. For example:

```
-> ipv6 address 4100:2::/64 eui-64 v6if-v300
```

Note. *Optional.* To verify the IPv6 interface configuration, enter **show ipv6 interface** For example:

```
-> show ipv6 interface
Name                               IPv6 Address/Prefix Length      Status  Device
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
v6if-v200                          fe80::2d0:95ff:fe12:fab5/64      Down    VLAN 200
                                      4100:1::2d0:95ff:fe12:fab5/64
                                      4100:1::/64
v6if-v300                          fe80::2d0:95ff:fe12:fab6/64      Down    VLAN 300
                                      4100:2::2d0:95ff:fe12:fab6/64
                                      4100:2::/64
loopback                          ::1/128                          Active  Loopback
                                      fe80::1/64
```

Note that the link-local addresses for the two new interfaces and the loopback interface were automatically created and included in the **show ipv6 interface** display output. In addition, the subnet router anycast address that corresponds to the unicast address is also automatically generated for the interface.

5 Enable RIPng for the switch by using the **ipv6 load rip** command. For example:

```
-> ipv6 load rip
```

6 Create a RIPng interface for each of the IPv6 VLAN interfaces by using the **ipv6 rip interface** command. For example:

```
-> ipv6 rip interface v6if-v200  
-> ipv6 rip interface v6if-v300
```

IPv6 routing is now configured for VLAN 200 and VLAN 300 interfaces, but it is not active until at least one port in each VLAN goes active.

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 29-5](#) 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 29-7](#) 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
```


Auto Configuration of IPv6 Addresses

This implementation of IPv6 supports the *stateless* auto configuration 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 auto configuration 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 29-11](#) for more information.

Both stateless and *stateful* auto configuration 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 auto configuration 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 auto configuration and IPv6 address notation.

Router Advertisement (RA) Filtering

RA filtering can be used to prevent the spread of rogue RAs from unauthorized systems. When enabled on an IPv6 VLAN, any received RAs will be dropped without being forwarded on to any other connected IPv6 clients.

One or more trusted ports or linkaggs can be specified for the VLAN. RAs received on those trusted ports or linkaggs will be allowed to the other IPv6 clients reached through the VLAN. See [“Configuring Router Advertisement \(RA\) Filtering” on page 29-20](#) “on page 29-20 for more information.

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 AOS Release 6 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 [“Auto Configuration of IPv6 Addresses” on page 29-7.](#)
- 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 AOS Release 6 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 29-5 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 RIPv6 (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 RIPv6 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
  RIPv6         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 29-14](#).
- 2 Configure redistribution to apply a route map, as described in [“Configuring Route Map Redistribution” on page 29-18](#).

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 AOS Release 6 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 29-18 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 AOS Release 6 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 29-14 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 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 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.

Configuring Router Advertisement (RA) Filtering

When RA filtering is enabled on a VLAN, router advertisements received on any port or linkagg are discarded. If one or more trusted ports or linkaggs are configured, RAs received on them will be accepted and sent on to any connected IPv6 nodes.

To enable RA filtering on a VLAN, use the **ipv6 ra-filter** command. For example:

```
-> ipv6 ra-filter vlan 5
```

This enables RA filtering on VLAN 5. All RAs received on the VLAN will be dropped.

To specify a trusted port or linkagg, specify the port or linkagg using the **ipv6 ra-filter** command with the 'trusted-port' option. For. For example:

```
-> ipv6 ra-filter vlan 5 trusted-port 1/22
```

This specifies that port 1/22 is trusted port in VLAN 5. RAs received on this port will be forwarded to all other clients connected through the VLAN. RAs received on any other port will still be dropped.

To remove a trusted port, use the following command. This will remove port 2 as a trusted port on VLAN 5.

```
-> no ipv6 ra-filter vlan 5 trusted-port 1/22
```

To disable RA filtering on a VLAN, use the **no ipv6 ra-filter** command. For example:

```
-> no ipv6 ra-filter vlan 5
```

This disables RA filtering on VLAN 5.

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. |
| show ipv6 ra-filter vlan | Displays the list of VLANs configured for RA filtering. |
| show ipv6 ra-filter counters | Displays the counter statistics of the NIs which are up. |

For more information about the displays that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

30 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 AOS Release 6 CLI Reference Guide*.

This chapter provides an overview of RIP and includes information about the following procedures:

- RIP Routing
 - Loading RIP (see [page 30-7](#))
 - Enabling RIP (see [page 30-7](#))
 - Creating a RIP Interface (see [page 30-7](#))
 - Enabling a RIP Interface (see [page 30-8](#))
- RIP Options
 - Configuring the RIP Forced Hold-Down Interval (see [page 30-10](#))
 - Configuring the RIP Update Interval (see [page 30-10](#))
 - Configuring the RIP Invalid Timer (see [page 30-10](#))
 - Configuring the RIP Garbage Timer (see [page 30-11](#))
 - Configuring the RIP Hold-Down Timer (see [page 30-11](#))
 - Enabling a RIP Host Route (see [page 30-11](#))
- RIP Redistribution
 - Configuring Route Redistribution (see [page 30-12](#))
- RIP Security
 - Configuring Authentication Type (see [page 30-18](#))
 - Configuring Passwords (see [page 30-18](#))

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 6350, 6450 |
| Maximum Number of RIP Peers | 10 |
| Maximum Number of RIP Interfaces | 10 |
| Maximum Number of RIP Routes | 256 |
| Maximum number of ECMP gateways | 4 |

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'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 30-6](#).

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.

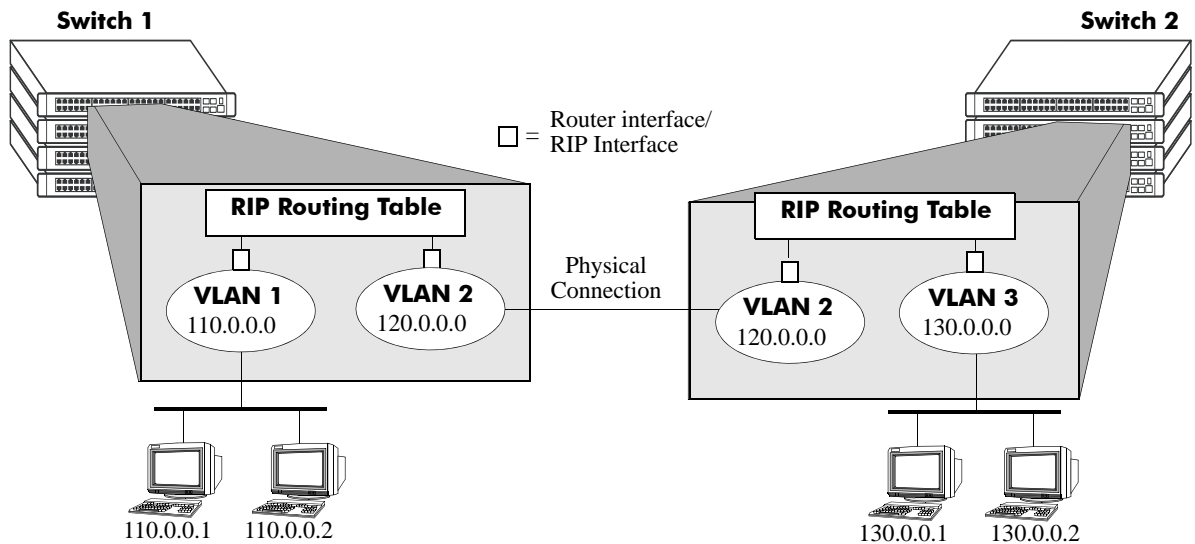
RIP Version 2

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 30-18](#).
- **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 rcv-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 rcv-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 30-11](#).

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 30-12](#).
- 2 Configure redistribution to apply a route map, as described in [“Configuring Route Map Redistribution” on page 30-16](#).

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 AOS Release 6 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 30-16](#) 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 AOS Release 6 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 30-12](#) 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

-> 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 AOS Release 6 CLI Reference Guide*.

31 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 AOS Release 6 CLI Reference Guide*.

The following procedures are described:

- [“Enabling/Disabling RDP” on page 31-8.](#)
- [“Creating an RDP Interface” on page 31-8.](#)
- [“Specifying an Advertisement Destination Address” on page 31-9.](#)
- [“Defining the Advertisement Interval” on page 31-9.](#)
- [“Setting the Advertisement Lifetime” on page 31-10.](#)
- [“Setting the Preference Levels for Router IP Addresses” on page 31-10.](#)
- [“Verifying the RDP Configuration” on page 31-11.](#)

RDP Specifications

| | |
|---|--|
| RFCs Supported | RFC 1256–ICMP Router Discovery Messages |
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 AOS Release 6 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 31-8.

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 AOS Release 6 CLI Reference Guide*.

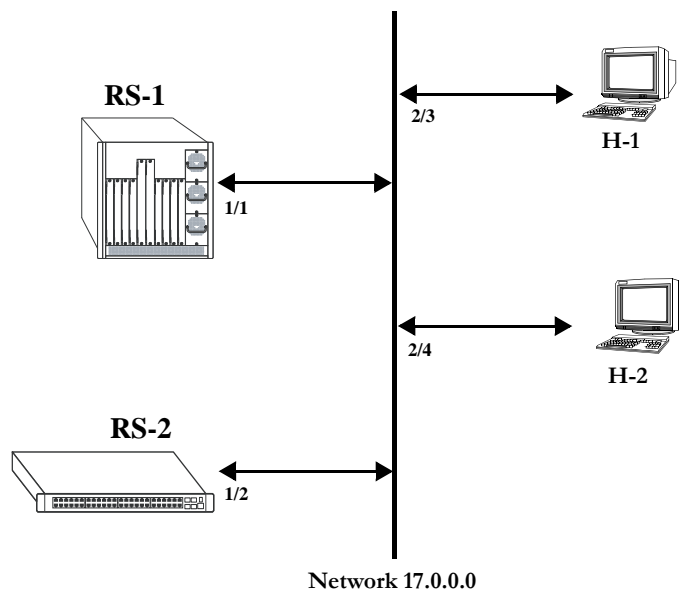
RDP Overview

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 31-9](#) 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 31-9](#) and [“Setting the Advertisement Lifetime” on page 31-10](#) 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 28, "Configuring IP,"](#) and [Chapter 39, "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 31-6](#) 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 28, “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 AOS Release 6 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 31-3](#).

32 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Quick steps for configuring DHCP Relay on [page 32-7](#).
- Setting the IP address for global DHCP on [page 32-14](#).
- Identifying the VLAN for per-VLAN DHCP on [page 32-15](#).
- Enabling BOOTP/DHCP Relay on [page 32-15](#).
- Setting the forward delay time on [page 32-16](#).
- Setting the maximum hops value on [page 32-16](#).
- Setting the relay forwarding option to standard or Per-VLAN on [page 32-16](#).
- Configuring the DHCP client interface to obtain an ip address for the switch on [page 32-17](#).
- Vendor Class Identifier and Preference to OXO DHCP Server [page 32-21](#)
- Configuring relay for generic UDP service ports on [page 32-21](#).
- Using the relay agent information option (Option-82) on [page 32-25](#).
- Using DHCP snooping on [page 32-28](#).

The different sections describing the DHCPv6 Relay functionality in this chapter are as follows:

- [“Quick Steps for Setting Up DHCPv6 Relay”](#) on page 32-8
- [“DHCPv6 Relay Overview”](#) on page 32-42
- [“Configuring DHCPv6 Relay”](#) on page 32-43
- [“Verifying the DHCPv6 Relay Configuration”](#) on page 32-52

For information about the IP protocol, see [Chapter 28, “Configuring IP.”](#) and IPv6 protocol see [Chapter 15, “IPv6 Commands”](#).

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 6350, 6450 |
| DHCP Relay Implementation | Global DHCP Per-VLAN DHCP Multiple VLAN tagging |
| DHCP Relay Service | BOOTP/DHCP (Bootstrap Protocol/Dynamic Host Configuration Protocol) |
| UDP Port Numbers | 67 for Request 68 for Response Multiple VLAN Tagging |
| IP address allocation mechanisms | Dynamic –DHCP assigns an IP address to a host for a limited period (or until the host explicitly relinquishes the address). Manual –The network administrator assigns an IP address of a host 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 eight 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 |
| Maximum number of DHCP Snooping VLANs | 64 |
| Maximum number of VLANs Supporting IP Source Filtering | 32 |
| Maximum number of clients per switching ASIC when IP source filtering is enabled. | OmniSwitch 6350 - 96 Clients OmniSwitch 6450 - 24 ports (256 Clients is supported) and 48 port (512 clients is supported; Port 1-24: 256 Entries Port 25-48: 256 Entries), when ip helper dhcp-snooping ip-source-filter is enabled. Note: If the client port is a linkagg port, then the number of the ports part of the linkagg will be considered as client ports and will affect the Maximum number of clients per switching ASIC when IP source filtering is enabled. |

DHCPv6 Relay Specifications

| | |
|---|---|
| RFCs Supported | RFC 3315 - Dynamic Host Configuration Protocol for IPv6 (DHCPv6) RFC 6221 - Lightweight DHCPv6 Relay Agent RFC 4649 - Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay agent Remote-ID option |
| Platforms Supported | OmniSwitch 6350, 6450 |
| DHCPv6 Relay Implementation | Global DHCP Per-VLAN DHCP |
| DHCPv6 Relay Service | DHCPv6 |
| DHCPv6 Implementation | VRF- On default VRF only |
| DHCPv6 LDRA | Per-VLAN Global DHCP |
| UDP Port Numbers for DHCPv6 Relay | 546 for Request 547 for Response |
| IPv6 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). |
| Maximum IPv6 addresses supported for each Relay Service | 256 IPv6 addresses for each Relay Service. |
| Global Relay | Up to 256 configurable IPv6 relay addresses |
| IPv6 addresses supported for the Per-VLAN service | Maximum of 8 IPv6 addresses for each VLAN relay service. Maximum of 256 VLAN relay services. |
| Maximum number of DHCPv6 Client interfaces | 1 |
| Maximum number of DHCPv6 Snooping VLANs | 64 |

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 |
| IP helper DHCP Snooping trap-mode setting | ip helper dhcp-snooping trap-mode | Default |

DHCPv6 Relay Defaults

The following table describes the default values of the DHCPv6 Relay parameters:

| Parameter Description | Command | Default Value/Comments |
|---|--|------------------------|
| Maximum number of hops | <code>ipv6 helper maximum hops</code> | 32 hops |
| Packet forwarding option | <code>ipv6 helper standard</code> <code>ipv6 helper per-vlan</code> | Standard |
| Link Aggregate level DHCPv6 Snooping Trust Mode | <code>ipv6 helper dhcp-snooping linkagg</code> | client-only-untrusted |
| Port-level DHCPv6 Snooping Trust Mode | <code>ipv6 helper dhcp-snooping port</code> | client-only-untrusted |
| VLAN-level DHCPv6 Snooping | <code>ip helper dhcp-snooping vlan</code> | Disabled |

Quick Steps for Setting Up DHCP Relay

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:

```
-> 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 on the show command, see the “DHCP Relay” chapter in the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Quick Steps for Setting Up DHCPv6 Relay

Configure DHCPv6 Relay on switches where packets are routed between IP networks.

There is no separate command for enabling or disabling the relay service. DHCPv6 Relay is automatically enabled on the switch whenever a DHCPv6 server IP address is defined. To set up DHCPv6 Relay, proceed as follows:

1 Identify the IP address of the DHCPv6 server. Where the DHCPv6 server has IP address 4100:1::0, use the following command:

```
-> ipv6 helper address 4100:1::0
```

2 Set the maximum hops count for the DHCPv6 relay. To set the hop count to 32, use the following command:

```
-> ipv6 helper maximum hops 32
```

Note. Optional. To verify the DHCPv6 Relay configuration, enter the [show ipv6 helper](#) command. The display shown for the DHCPv6 Relay configured in the above Quick Steps is shown here:

```
-> show ipv6 helper
Dhcpv6 helper :
  Max number of hops          = 32,
  DHCPV6 Snooping Status      = VLAN-Level Enabled,
  DHCPV6 Remote-id            = Enabled,
  DHCPV6 Remote-id Enterprise Number = -,
  DHCPV6 Interface-id Prefix  = -,
  DHCPV6 Snooping Binding DB Status = Enabled,
  Database Sync Timeout       = 300,
  Database Last Sync Time     = Jun 1 2015 15:17,
  Binding Persistency Status  = Disabled,
  Forward option              = per-vlan only
```

```
Forwarding Address          Vlan Number
-----+-----
2001:1:2000::4             2002
```

For more information about this display, see the “Configuring IPv6” chapter in the *OmniSwitch AOS Release 6 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 verifies if the forward delay time (specified by the user) has elapsed and then sends 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 is sent to all the 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 is discarded by the DHCP Relay.

The forwarding Option allows you to specify if the relay must 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 32-16](#) for more information.

The relay function can also 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 allocate reusable network addresses and additional configuration options automatically. 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 (or until the host explicitly relinquishes the address).

Manual—The network administrator assigns an IP address of a host and DHCP simply conveys the assigned address to the host.

DHCP and the OmniSwitch

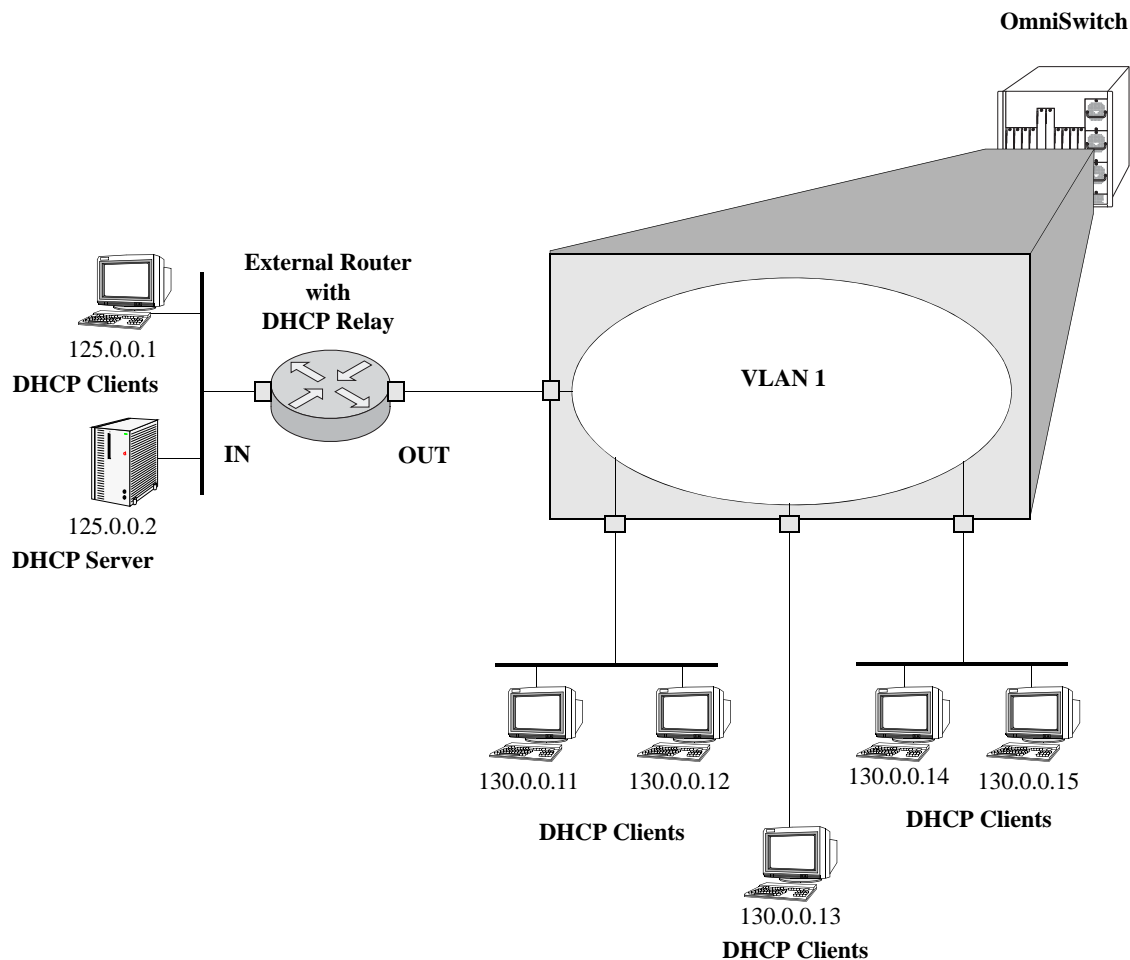
The unique characteristics of the DHCP protocol require a good plan before setting up the switch in a DHCP environment. Since DHCP clients initially have no IP address, placement of these clients in a VLAN is hard to determine. In simple networks (for example, one VLAN), rules need not be deployed to support the BOOTP/DHCP relay functionality.

In multiple VLAN network configurations, VLAN rules can be deployed to support the processing and relay of DHCP packets strategically. 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 9, “Defining VLAN Rules,”](#) for more information.

External DHCP Relay Application

The DHCP Relay can 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 that 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.



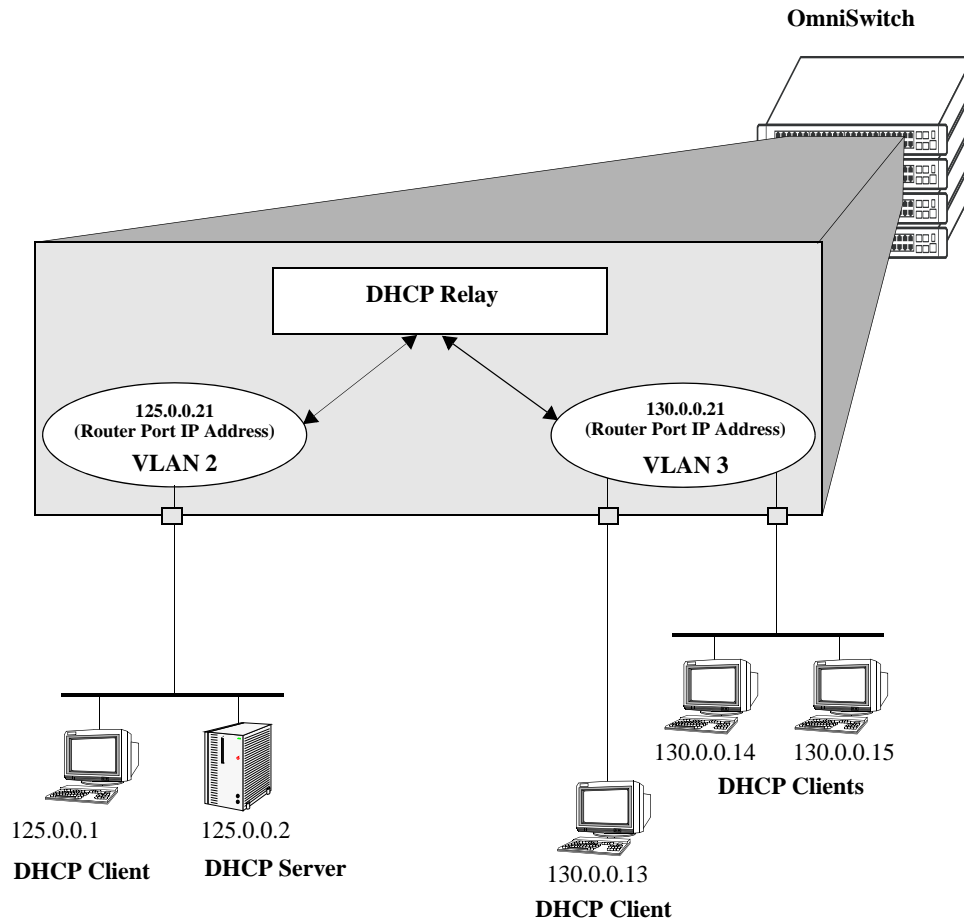
DHCP Clients are Members of the Same VLAN

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 have the same subnet address (130.0.0.0) inserted into each of the requests by the DHCP Relay function of the router. The DHCP server assigns 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 32-13.

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 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 9, “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 through the DHCP Relay routing entity in the switch. When a DHCP request frame is received by the DHCP Relay entity, it is 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. Set up a global DHCP request or 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, 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. 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. A maximum of 256 addresses can be configured 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 is 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, identify the number of the VLAN that makes the relay request.

Identifying the VLAN

Enter one or more server IP addresses to which packets are sent from a specified VLAN by using the **ip helper address vlan** command. The following syntax identifies 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 is 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 is defined and the DHCP Relay is configured for either Global DHCP request or Per-VLAN DHCP request, 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.

The relay function can also 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 time, measured in seconds, since the client last booted. DHCP Relay will not process the packet unless the elapsed boot time value of the client 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 seconds 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 ignores this value.

Setting the Relay Forwarding Option

This value specifies if DHCP Relay must 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 obtain an IP address dynamically 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-2 is supported for configuring the time zone.
- 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 is released. When not specified by the DHCP server, a default lease time of seven days is allocated.
- The system name and the time zone of the OmniSwitch is set according to the system name (DHCP Option-12) and time zone (DHCP Option-2) assigned by the DHCP server. However, if user configures the system name and time zone to non-default values, DHCP server does not assign the system name and time zone values.
- The DHCP Client-enabled IP address serves as the primary IP address when multiple addresses are configured for a VLAN.

DHCP Option-12 and DHCP Option-2

Some points to note:

- DHCP server sets the Option-2 and Option-12 values only when they are set to their default values ("GMT" is default value for time zone and "vxTarget" is default value for system name) or they are already set by the DHCP. Once the user configures these values to non-default values, DHCP does not set them.
- The user-defined configuration (through CLI, WebView, SNMP) for system name and time zone gets priority over the DHCP server values. To re-enable to DHCP mode, user has to configure the system name and time zone explicitly to their default values (if set to non-default values).

Note. User-defined values for the system name and time zone can be set using [system name](#) and [system timezone](#) CLI commands.

- Periodic DHCPINFORM message is sent to the DHCP server every ten minutes requesting for Option-2 and Option-12 (only after successfully acquiring the DHCP lease from the server). The DHCP server values are compared to the existing time zone and system name values, and the values are applied only if there is a change in value and the user have not configured them. This helps in applying the changes done in the DHCP server on the fly.

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 then tries to get a new IP address after the switch-assigned DHCP lease time expires. 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:

- 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 is 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.
- Ensure that 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 canceled.
- When a DHCP release is performed, the system name remains unchanged even if the name was updated using the DHCP client Option-12 information.
- Some IP phones require vendor specific DHCP options, namely the option 176 and 242. Such options are set by the admin with a specific string.

DHCP Server Preference in DHCP Client Interface

DHCP server preference for the DHCP client can be configured on the switch. This allows the DHCP client to accept the lease from the highest priority server from the multiple DHCP offers received.

When server-preference is enabled in the switch. The client receives multiple DHCP OFFER messages, the server-preference logic would determine which of the DHCP servers must be given priority and the IP address provided by that DHCP server will be accepted.

The type of DHCP server sending offers will be identified by the VSI string (option 43) configuration.

When server-preference is enabled, the following precedence order is followed:

- 1.OVCloud : "alenterprise"
- 2.OVClient:"alcatel.nms.ov2500"
- 3.OXO:"alcatel.a4400.0"
- 4.Others / Undesired :Identified by absence of VSI string

A 30 second time window is activated when DHCP client interface is created with server-preference enabled. First preference is given for the OVCloud server, the client waits for 30 seconds from the time of sending discovery even after receiving OFFER from other servers other than OVCloud. If the OFFER from the OVCloud server is not received in that 30 second time period the OFFER from the next priority server or other server is accepted.

Configuring DHCP Server Preference

The DHCP server preference can be enabled using the **ip-interface dhcp-client** command. Example,

```
-> ip interface dhcp-client vlan 1 server-preference
```

The server preference option can also be set without specifying VLAN ID, provided the dhcp-client interface is associated with a VLAN prior to setting the server preference. Example,

```
-> ip interface dhcp-client server-preference
```

Note. If server-preference option is enabled on a switch where dhcp-client has already obtained a lease, the obtained lease will be retained until the client triggers next DHCP discovery message.

The server-preference option is mutually exclusive with the vsi-accept-filter option. Switching from vsi-filter setting to server-preference option is allowed only after removing the existing vsi-accept-filter option by resetting it to default value “”. Likewise, switching from server-preference option to vsi-accept-filter setting is allowed only after removing the existing preference using 'no' keyword.

The DHCP server preference can be removed using the **no** form of the command. Example,

```
-> ip interface dhcp-client no server-preference
```

The configured DHCP server preference option can be viewed using the **show ip interface dhcp-client** command.

Vendor Class Identifier and Preference to OXO DHCP Server

DHCP Vendor Class and Switch Type

The information of vendor class and switch type will be sent in DHCP discover or request packets during the Remote Configuration Load (RCL). The vendor class and switch type will be a part of DHCP option-60 filed as OmniSwitch-<Module Type> (Example: OmniSwitch-OS6450-24). The OXO server will send its information in Option-43 in DHCP OFFER or ACK, and AOS will decoded Option-43 to give preference to the OXO server.

The DHCP client extracts the Vendor Specific Information (VSI) from the DHCP response packets (DHCPACK) to provide preference to the desired OXO DHCP server. The response from the OXO server is recognized with its VSI (Example: "alcatel.a4400.0").

The VSI string "alcatel.a4400.0" is hard coded and is used to configure the OXO server during the RCL process. To configure the OXO server after the RCL process, VSI filter is used.

Note. The string "alcatel.a4400.0" is hard coded. This will be used to match with the extracted vendor specific information from DHCP response during the RCL process.
The OXO DHCP server can be changed only after the RCL process by configuring the desired OXO server address in the VSI filter.

Configuring DHCP client for preferred OXO DHCP Server

In order to have consistency in operation, the switch should use the same IP address as originally given by the OXO during RCL. To filter the DHCP response from non-DHCP and OXO DHCP server after RCL, use **ip interface dhcp-client** command. For example, the following command sets the preference to the OXO DHCP server having the vendor class ID "alcatel.a4400.0" on default VLAN 1:

```
-> ip interface dhcp-client vlan 1 ifindex 1 vsi-accept-filter "alcatel.a4400.0"
```

Note. The OXO DHCP server response will be sent only on default VLAN 1.
In order to retain the same OXO server which was configured before RCL, the VSI filter must match the hard coded string configuration "alcatel.a4400.0".

To view the configured vsi-accept-filter, use the **show ip interface** command.

For more information about RCL process and OXO preference mechanism, see the "Managing Automatic Remote Configuration Download" chapter in the *OmniSwitch AOS Release 6 Switch Management Guide*.

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 (that is, 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 known ports 67 and 68.
- If the BOOTP/DHCP relay service is disabled, the **ip helper** configuration is *not* retained and all dependant functionality (that is, 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 in the following section.

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 AOS Release 6 CLI Reference Guide*.

Specifying a Forwarding VLAN

To specify which VLANs 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
```

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 AOS Release 6 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.

Note. DHCP Snooping now provides multiple VLAN tagging support.

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 **ip helper dhcp-snooping option-82 format** command is used to configure the type of data (base MAC address, system name, interface alias, or user-defined) that is inserted into the above Option-82 suboptions. The system name and user-defined text are reported in ASCII text format, but the MAC address is still reported in hex-based format.

By default, the relay agent drops client DHCP packets it receives that already contain Option-82 data. However, it is possible to configure an Option-82 policy to specify how such packets are treated. See [“Configuring a Relay Agent Information Option-82 Policy” on page 24-18](#) for more information.

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

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 retains the Option-82 data received and sends 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 drops 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 drops 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
```

DHCP Option-82 functionality is not restricted to ports associated with a specific VLAN as this feature is not available on a per-VLAN basis. 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 described, 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
```

DHCP Option-82 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 drops 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.

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 32-38](#) for more information.
- **IP Source Filtering**—Restricts DHCP Snooping port traffic to only packets that contain the client IP address, MAC address, port, and VLAN combination obtained from the DHCP snooping binding table entry. The DHCP Snooping binding table is used to verify the client information for the port that is enabled for IP source filtering. See [“Configuring IP Source Filtering” on page 32-34](#) for more information.
- **DHCP Snooping Global Mode Settings**—Globally configures the DHCP Snooping trap-mode settings for the switch. See [“DHCP Snooping Global Mode Settings” on page 32-38](#) 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 39, “Configuring QoS,”](#) in the *OmniSwitch AOS Release 6 Network Configuration Guide* for more information.

When DHCP Snooping is first enabled, all ports are considered untrusted. It is important to configure ports connected to a DHCP server inside the network as trusted ports. See [“Configuring the Port Trust Mode” on page 32-32](#) 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 drops 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 32-32](#) 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 32-15](#) 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 32-38](#) 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 32-29](#) for more information.
- Configure ports connected to DHCP servers within the network as trusted ports. See [“Configuring the Port Trust Mode” on page 32-32](#) for more information.
- Ensure that Option-82 data insertion is always enabled at the switch or VLAN level. See [“Enabling DHCP Snooping” on page 32-29](#) 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 32-32](#) 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 (for example, ports 1–24, 25–48, and so on) 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 32-25](#) 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 32-32](#) 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 suboptions 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:

- If the binding table is enabled, disabling Option-82 is not allowed.
- If Option-82 data insertion is not enabled at either the switch or VLAN level, enabling the binding table is not allowed.

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. 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
```

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 32-36](#) 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 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
```

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 ip-source-filter** command to enable or disable this function for a specific port or range of ports. For example:

```
-> ip helper dhcp-snooping ip-source-filtering port 1/10 enable
-> ip helper dhcp-snooping ip-source-filtering port 2/1-5 enable
```

When IP source filtering is enabled, the maximum number of clients supported is 125 per switching ASIC. Each switching ASIC controls 24 ports (for example, ports 1–24, 25–48, and so on) on a module.

Note. The command **ip helper dhcp-snooping port ip-source-filter** is deprecated from the release 6.6.4. The command is supported for backward compatibility. The command **ip helper dhcp-snooping ip-source-filter** is used to configure IP source filtering.

IP source filtering applies to DHCP Snooping ports and restricts port traffic to only packets that contain the proper client source information in the packet. The DHCP Snooping binding table is used to verify the client information for the port that is enabled for IP source filtering.

Port Source Filtering -Filters based on interface number, source MAC-address and source IP address.

By default IP source filtering is disabled for a DHCP Snooping port. Use the **ip helper dhcp-snooping ip-source-filter** command to enable or disable this function.

For example, to enable source filtering on individual port 1/1, enter:

```
-> ip helper dhcp-snooping ip-source-filter port 1/1 enable
```

To enable source filtering on link aggregate 2, enter:

```
-> ip helper dhcp-snooping ip-source-filter linkagg 2 enable
```

Configuring IP Source Filtering

IP source filtering uses the binding table information to protect the network from spoofing attacks. The binding table information is built by DHCP transactions between the clients connected to the IP source filtering enabled switch and the server.

IP source filtering capability can be enabled or disabled at a port, link aggregation, or VLAN level. When this function is enabled, the switch allows the traffic that matches the client IP address, MAC address, port, and VLAN combination obtained from the DHCP snooping binding table entry. All the other packets will be dropped by default.

IP source filtering can be bypassed on specific subnets on VLAN level. When this feature is configured, all the packets ingressing through the IP source filtering enabled port/VLAN are checked against the exceptional subnet entries in the hardware. Only the packets which are in-sync with the binding information are allowed, the rest of the traffic is dropped. So, only the clients having valid exceptional subnet entry and binding information would be able to send traffic through the switch. The exceptional subnet entry consists of IP subnet and subnet mask.

Note.

- Source filtering can be enabled only on the VLANs on which the DHCP Snooping is enabled.
 - Source filtering can be enabled
 - on the ports that are associated with a VLAN on which DHCP Snooping is enabled.
 - on all the ports when DHCP Snooping is globally enabled for the switch.
-

To enable the IP source filtering capability at a port, link aggregation, or VLAN level, use the **ip helper dhcp-snooping ip-source-filter** command. By default, IP source filtering is disabled for a port or link aggregate, or VLAN.

For example, to enable source filtering on individual port 1/1, enter:

```
-> ip helper dhcp-snooping ip-source-filter port 1/1 enable
```

To enable source filtering on link aggregate 2, enter:

```
-> ip helper dhcp-snooping ip-source-filter linkagg 2 enable
```

To enable source filtering on VLAN 10, enter:

```
-> ip helper dhcp-snooping ip-source-filter vlan 10 enable
```

To disable source filtering on VLAN 10, enter:

```
-> ip helper dhcp-snooping ip-source-filter vlan 10 disable
```

Enable or Disable Exceptional Subnets Over Source Filtering

To enable or disable specific subnets on VLAN level from IP source filtering, configure the required subnets using the **ip helper dhcp-snooping ip-source-filter** command. The specified subnets will be excluded from IP source filtering.

```
-> ip helper dhcp-snooping ip-source-filter vlan 4050 allow 10.55.40.4 mask  
255.255.255.252 enable
```

```
-> ip helper dhcp-snooping ip-source-filter vlan 4050 allow 10.55.40.4 mask  
255.255.255.252 disable
```

Note. A maximum of 16 subnets are allowed to be excluded from source filtering on OmniSwitch 6450. On OmniSwitch 6350, only 8 subnets are allowed.

show ip helper dhcp-snooping ip-source-filter vlan displays the subnet information on which IP source filtering is excluded.

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
```

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 AOS Release 6 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. 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 32-36 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 appear 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.

However when the DHCP client host is connected to another port, the associated binding table is moved from the previous port to the new port.

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.

DHCP Snooping ISF ARP-Allow

By default ARP packets are checked against binding entries and are allowed only when a valid binding entry for that client on the port is already present. By enabling this feature, ARP packets are not checked against the binding entries and are allowed to pass through transparently.

For example, consider a client device connected to a snooping device on port A of the switch. Now, if the client device is disconnected from port A and connected to port B of the same switch, an ARP request is sent to the client. The client receives the ARP request and generates the ARP reply, this ARP reply will be dropped by the ISF because there is no binding entry for this client on the new port B.

To avoid the ARP reply packets being dropped from the clients, DHCP Snooping IP source filter ARP-allow function can be configured.

To enable the DHCP Snooping ISF ARP-allow function, use the **ip helper dhcp-snooping ip-source-filter arp-allow** command. For example:

```
-> ip helper dhcp-snooping ip-source-filter arp-allow enable
```

DHCP Snooping and ISF must be enabled before enabling this function. On enabling this feature an entry is made in the ISF table hence the number of binding entry is reduced by one.

To disable the DHCP Snooping ISF ARP-allow function, enter:

```
-> ip helper dhcp-snooping ip-source-filter arp-allow disable
```

Use the **show ip helper** command to view the operational status of the DHCP Snooping ISF ARP allow function.

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.

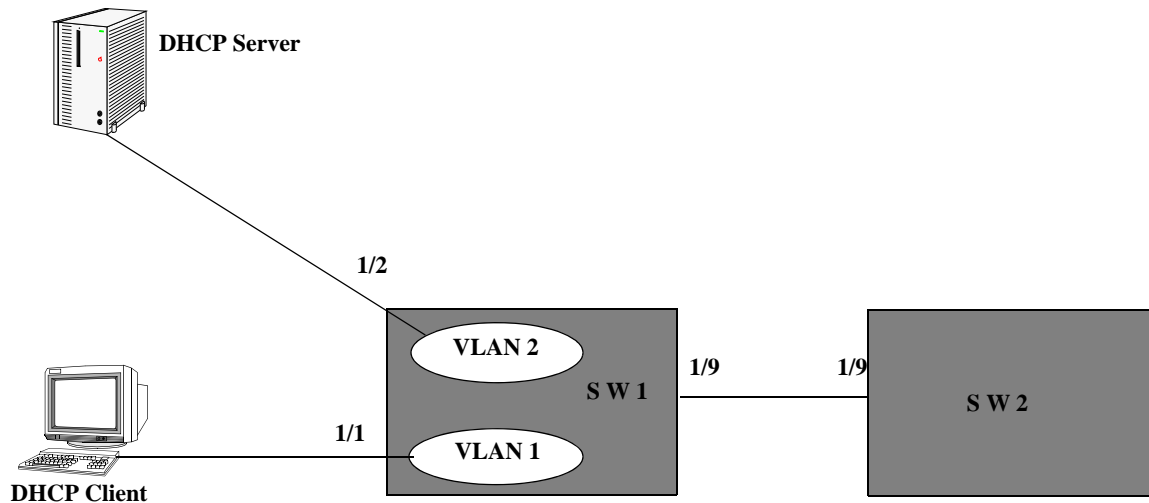
DHCP Snooping Global Mode Settings

When DHCP snooping is globally enabled, the DHCP packets with unicast MAC and unicast IP are forwarded based on the MAC entry information previously cached. Hence, the packets that are received on the DHCP snooping switch in client VLAN will not be dropped as there is a MAC present in the DHCP snooping table.

In a scenario where the DHCP client and the DHCP server are on different VLANs on the same switch and a relay agent is used to process the DHCP packets. The DHCP snooping binding entry is overwritten when the DHCP packets are relayed back to the switch from the DHCP relay agent on a trusted port. This overwriting of the binding entry will cause the DHCP server packets to be dropped when received on the client VLAN due to the mismatch in the binding entry.

Hence, when globaluturn is enabled on the switch the binding table will not be built for the unicast BOOTP packets sent on the trusted ports. So the binding table entry is not modified when the DHCP packet is relayed to the switch from the relay agent and the DHCP server packets are not dropped when received on the client VLAN.

An example of the scenario is shown in the following section:



The DHCP client connected to the Switch 1 (SW1) uses the untrusted port 1/1 on VLAN 1 to send the DHCP client packets. The switch makes a binding table entry and forwards the packet to the DHCP relay agent (SW2) through the trusted port 1/9. The relay agent resends the packet back to the switch (SW1) on which the DHCP server is configured on VLAN 2 on trusted port 1/2. The switch (SW1) overwrites the binding table entry for the packet sent on the trusted port. When the DHCP server responds, the DHCP server packets are dropped when received on the client VLAN since the binding table is overwritten with the trusted port information.

When Globaluturn is enabled on the switch the binding table will not be built for the unicast BOOTP packets sent on the trusted ports, hence avoiding the overwrite of the binding table entry and dropping of the DHCP server packets on the client VLAN.

Configuring the DHCP Snooping Global Mode Settings

To configure the DHCP snooping global mode settings, use the [ip helper dhcp-snooping trap-mode](#) command.

There are four setting option for the global mode:

- **default:** Default DHCP snooping functionality is followed. Only the source packets are trapped to the CPU.
- **reverse-enable:** Disables DHCP binding entry on a trusted port. The binding table will not be built for the unicast BOOTP packets sent on the trusted ports.
- **hardware:** The packets sent with source port 67 and destination port 67 will not be trapped to the CPU during DHCP snooping. In this mode there are chances of the packet with 67/67 being spoofed and the DHCP snooping can be by-passed.
- **software:** The DHCP packets with 67/67 pair is trapped to the CPU in addition to 67/68 and 68/67. This mode can lead to CPU spikes. Since all the DHCP packets irrespective of port pair is trapped to the CPU.

For example:

To enable globaluturn setting, use the following command:

```
-> ip helper dhcp-snooping trap-mode reverse-enable
```

For more information on using the CLI, See the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Use the **show ip helper** command to verify the trap-mode of the DHCP snooping global mode setting.

Note.

The global DHCP snooping settings will not work:

- If the DHCP server message such as OFFER, NAK, and ACK are received on untrusted ports such as client only or blocked ports.
 - If the client VLAN is a plain L2 network, that is either source MAC or the destination MAC is same as the BOOTP MAC.
-

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 AOS Release 6 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 32-7.

| | |
|--|---|
| 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 configured for that service, and the number of packets the service has sent and received. |
| show ip helper dhcp-snooping ip-source-filter | Displays the ports or VLANs on which IP source filtering is enabled or the binding table for IP source filtering enabled ports. |
| show ip helper dhcp-snooping vlan | Displays a list of VLANs that have DHCP Snooping enabled and whether MAC address verification and Option-82 data insertion are 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). |

DHCPv6 Relay Overview

The Alcatel-Lucent OmniSwitch implementation of RFC 3315 contains IPv6 support and provides stateless address auto configurations to IPv6 hosts connected to the switch.

Every IPv6 host is assigned with a global IPv6 address either in Stateless or Stateful mode. This is decided by IPv6 router located on the network. IPv6 router sends out Router Advertisement (RA) multicast messages periodically using the address ff02::1. IPv6 hosts upon boot up process this RA message to decide its address configuration mode.

DHCPv6 is used to acquire global IPv6 address in Stateful mode and DHCPv6 messages are exchanged between IPv6 hosts and IPv6 router similar to client-server model. The IPv6 addresses are assigned by DHCPv6 server in Stateful mode. The DHCPv6 server maintains the client information.

All DHCPv6 messages triggered by DHCPv6 clients are processed by AOS switch through DHCPv6 relay and are forwarded to the configured DHCPv6 relay agent as unicast packet.

DHCPv6 Relay on OmniSwitch processes and forwards all DHCPv6 messages triggered by DHCPv6 client to the configured DHCPv6 relay agent as a unicast packet.

Currently the following modes of DHCPv6 Relay are available:

- **DHCPv6 L3 Relay** - Switch acts as a pure Layer 3 relay agent when client facing interface has an IPv6 interface associated.
- **DHCPv6 LDRA** - Switch acts as a Lightweight DHCPv6 Relay Agent (LDRA) when client facing interface has no IPv6 interface and only VLAN is configured on it.

For details on how DHCPv6 Relay and configuration is implemented on OmniSwitch, see the following sections.

Configuring DHCPv6 Relay

The following section details the functionalities available and different CLI commands used for configuring DHCPv6 Relay.

Layer 3 DHCPv6 relay

The DHCPv6 Layer 3 Relay configuration has the following modes similar to DHCP relay

- **Global mode** - Up to 256 configurable IPv6 relay addresses.
- **Per-VLAN mode** - Up to 256 VLANs with up to 8 IPv6 relay addresses Per-VLAN.

This can be configured using the **ipv6 helper address** command family. For details on usage, see [“Configuring the DHCPv6 Snooping Binding Table” on page 32-49](#)

Layer 2 DHCPv6 relay or Lightweight DHCPv6 Relay Agent (LDRA)

The LDRA feature performs DHCPv6 snooping. The LDRA uses the following messages for relay-forwarding:

- **Relay-Forward**
 - The link-address is set to the unspecified address
 - The peer-address is copied from the client link local address
 - The Interface-ID option must be inserted
- **Relay-Reply**

Messages received on clients ports are only forwarded to trusted ports and not to other client ports. On client ports, the following messages are discarded as server violations:

- Advertise
- Reply
- Reconfigure
- Relay-Reply

A client port can also be configured as client-only-trusted or client-only-untrusted. When a client port is client-only-untrusted, the Relay-Forward message is discarded. The LDRA intercepts any DHCPv6 message received on client ports. DHCPv6 messages are identified with a source address, destination address for multicast as All DHCPv6 Relay Agent and Servers (FF02::1:2) and a UDP destination port 547.

Global DHCPv6

For the global DHCPv6 service, you must identify an IP address for the DHCPv6 server.

Setting the IPv6 Address

The DHCPv6 Relay is automatically enabled on a switch whenever a DHCPv6 server IP address is defined by using the **ipv6 helper address** command. There is no separate command for enabling or disabling the relay service. You should configure DHCPv6 Relay on switches where packets are routed between IP networks. The following command defines a DHCPv6 server address:

```
-> ipv6 helper address 2001::5
```

The DHCPv6 Relay forwards DHCPv6 broadcasts to and from the specified address. If multiple DHCPv6 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 IPv6 address, use the **no** form of the **ipv6 helper address** command. The IP address specified with this syntax is deleted. If an IP address is not specified with this syntax, then *all* IPv6 helper addresses are deleted. This is not applicable for per VLAN mode.

The following command deletes an IP helper address:

```
-> ipv6 helper no address 2001::5
```

Per-VLAN DHCPv6

For the Per-VLAN DHCPv6 service, you must identify the number of the VLAN that makes the relay request.

Identifying the VLAN

You can enter one or more server IPv6 addresses to which packets can be sent from a specified VLAN. Do this by using the **ipv6 helper vlan** command. The following syntax identifies the IPv6 address 2001::5 as the DHCPv6 server for VLAN 100:

```
-> ipv6 helper address 2001::5 vlan 100
```

The following syntax identifies the IPv6 address 2001::5 as the DHCPv6 server for range of VLANs from 100 to 105:

```
-> ipv6 helper address 2001::5 vlan 100-105
```

To delete an IPv6 address, use the **no** form of the **ipv6 helper address** command. The IPv6 address specified with this syntax is deleted. If an IPv6 address is not specified with this syntax, then *all* IPv6 helper addresses are deleted. The following command deletes a helper address for IPv6 address 2001::5 for the specified VLAN:

```
-> ipv6 helper no address 2001::5 vlan 200
```

Configuring DHCPv6 Relay Parameters

Once the IPv6 address of the DHCPv6 server(s) is defined and the DHCPv6 Relay is configured for either Global DHCPv6 request or Per-VLAN DHCPv6 request, you can set the following optional parameter values to configure DHCPv6 relay.

- The hop count
- The relay forwarding option

The only parameter that is required for DHCPv6 relay is the IPv6 address to the DHCPv6 server. The default values can be accepted for hop count and relay forwarding option.

Alternately the relay function can be provided by an external router connected to the switch; in this case, the relay is configured on the external router.

Setting Maximum Hops

This value specifies the maximum number of relays the DHCPv6 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, DHCPv6 Relay discards the packet. The following syntax is used to set a maximum of four hops:

```
-> ipv6 helper maximum hops 4
```

The hops value represents the maximum number of relays. The range is from one to 32 hops. The default maximum hops value is set to 32. This maximum hops value only applies to DHCPv6 Relay. All other switch services ignore this value.

Setting the DHCPv6 Relay Forwarding Option

This value specifies if DHCPv6 Relay must 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 **ipv6 helper** followed by **standard** or **per-vlan only** options. For example:

```
-> ipv6 helper standard  
-> ipv6 helper per-vlan only
```

Using the DHCPv6 Relay Agent Information

This implementation of the DHCPv6 relay agent information feature is based on the functionality defined Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay agent Remote-ID option RFC 4649. The **ipv6 helper remote id format** command is used to enable this feature at the switch level.

When this feature is enabled, communications between a DHCPv6 client and a DHCPv6 server are authenticated by the relay agent. The agent fills in the following information by default for each of these sub options:

- **Interface ID**—By default, the VLAN ID and **slot/port** from where the DHCPv6 packet originated.
- **Remote ID**— By default, configure the **enterprise number** and then the **remote ID**.

Interface ID and Remote ID can both be configured by the user.

Configuring Interface ID

Use the **ipv6 helper interface-id prefix** command to configure Interface ID manually with a user-defined string. For example,

```
-> ipv6 helper interface-id prefix pool-1
```

To disable or remove the **interface-id** prefix use the **no** option as follows:

```
-> ipv6 helper no interface-id prefix
```

Configuring Remote ID

Use the **ipv6 helper remote-id format** command to configures the type of information that is inserted into the Remote ID sub option. The information is inserted into the Remote ID field in ASCII text string format.

Use the following commands to set the **remote-id** in the relevant formats. For details on syntax definition refer the **ipv6 helper remote-id format** command in *OmniSwitch AOS Release 6 CLI Reference Guide*.

```
-> ipv6 helper remote-id enterprise-number 100
```

```
-> ipv6 helper remote-id format base-mac
```

```
-> ipv6 helper remote-id format vlan
```

To configure the switch to use the **interface-alias** previously configured using the **interfaces alias** command:

```
-> ipv6 helper remote-id format interface-alias
```

To configure the switch to automatically generate the **interface-alias** in the system name and slot/port format use the following command:

```
-> ipv6 helper remote-id format auto-interface-alias
```

To disable **remote-id** or remove the **enterprise-number** use the **disable** option as follows:

```
-> ipv6 helper remote-id format disable
```

VRF Support

The global relay IPv6 address or per-VLAN relay IPv6 address can only be configured on the default VRF.

DHCPv6 Snooping Configuration Guidelines

Consider the following when configuring the DHCPv6 Snooping feature:

- Layer 3 DHCPv6 Snooping requires the use of the relay agent to process DHCPv6 packets. As a result, DHCPv6 clients and servers must reside in different VLANs so that the relay agent is engaged to forward packets between the VLAN domains. See [“Configuring DHCPv6 Relay Parameters” on page 32-45](#) for information about how to configure the relay agent on the switch.
- Layer 2 DHCPv6 Snooping does not require the use of the relay agent to process DHCPv6 packets. As a result, an IPv6 interface is not needed for the client/server VLAN. See [“Layer 2 DHCPv6 relay or Lightweight DHCPv6 Relay Agent \(LDRA\)” on page 32-43](#) for more information.
- Both Layer 2 and Layer 3 DHCPv6 Snooping are active when DHCPv6 Snooping is globally enabled for the switch or enabled on a one or more VLANs. See [“Enabling DHCPv6 Snooping” on page 32-47](#) for more information.
- Configure ports connected to DHCPv6 servers within the network as trusted ports. See [“Configuring the Trust Mode for Ports and Link Aggregates” on page 32-48](#) for more information.

Enabling DHCPv6 Snooping

There are two levels of operation available for the DHCPv6 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.

Note. DHCPv6 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 DHCPv6 servers as trusted ports so that traffic to/from the server is not dropped.

Switch-level DHCPv6 Snooping

By default, DHCPv6 Snooping is disabled for the switch. To enable this feature at the switch level, use the **ipv6 helper dhcp-snooping** command. For example:

```
-> ipv6 helper dhcp-snooping enable
```

When DHCPv6 Snooping is enabled at the switch level, all DHCPv6 packets received on all switch ports and link aggregates are screened/filtered by DHCPv6 Snooping. By default, only client DHCPv6 traffic is allowed on the ports, unless the trust mode for a port is configured to block or allow all DHCPv6 traffic. See [“Configuring the Trust Mode for Ports and Link Aggregates” on page 32-48](#) for more information.

In addition, the following functionality is also activated by default when switch-level DHCPv6 Snooping is enabled:

- The DHCPv6 Snooping binding table is created and maintained. To configure the status of DHCPv6 snooping, use the **ipv6 helper dhcp-snooping binding** command.

VLAN-Level DHCPv6 Snooping

To enable DHCPv6 Snooping at the VLAN level, use the **ipv6 helper dhcp-snooping vlan** command. For example, the following command enables DHCPv6 Snooping for VLAN 200:

```
-> ipv6 helper dhcp-snooping vlan 200
```

When this feature is enabled at the VLAN level, DHCPv6 Snooping functionality is only applied to ports that are associated with a VLAN that has this feature enabled. Up to 256 VLANs can have DHCPv6 Snooping enabled. Note that enabling DHCPv6 Snooping at the switch level is not allowed if it is enabled for one or more VLANs.

Note. If DHCPv6 Snooping is *not* enabled for a VLAN, then all ports associated with the VLAN are considered trusted ports.

Configuring the Trust Mode for Ports and Link Aggregates

The DHCPv6 Snooping trust mode for a port or link aggregate determines whether or not the port or link aggregate accepts all DHCPv6 traffic, DHCPv6 traffic from IPv6 clients, or blocks all DHCPv6 traffic. The following trust modes are configurable using the **ipv6 helper dhcp port** and **ipv6 helper dhcp linkagg** commands:

- **client-only-untrusted**—The default mode applied to ports and link aggregates when DHCPv6 Snooping is enabled. Allows only DHCPv6 traffic from IPv6 clients on the ports with DHCPv6 snooping enabled. This mode restricts DHCPv6 traffic on the port or link aggregate to only DHCPv6 client-related traffic. When this mode is active for the port or link aggregate, the port or link aggregate is considered an untrusted interface. The relay forward message is not allowed along with the DHCPv6 messages
- **client-only-trusted**—This mode does not restrict DHCPv6 traffic on the port or link aggregate. When this mode is active on a port, the port is considered a trusted interface. In this mode the port or link aggregate behaves as if DHCPv6 Snooping is not enabled. The relay forward message is allowed when this mode is active.
- **trusted**—Allows all DHCPv6 traffic on the port or link aggregate. The port or link aggregate behaves as if DHCPv6 Snooping is not enabled.
- **block**—This mode blocks all DHCPv6 traffic on the port or link aggregate. When this mode is active for the port or link aggregate, the port or link aggregate is considered an untrusted interface.

To configure the trust mode for one or more ports, use the **ipv6 helper dhcp-snooping port** command. For example, the following command changes the trust mode for port 1/12 to blocked:

```
-> ipv6 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:

```
-> ipv6 helper dhcp-snooping port 2/1-10 trusted
```

Note. It is necessary to configure ports connected to DHCPv6 servers within the network and/or firewall as trusted ports so that necessary DHCPv6 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.

Similarly, to configure the trust mode for link aggregates, use the **ipv6 helper dhcp-snooping linkagg** command. For example,

```
-> ipv6 helper dhcp-snooping linkagg 1 trusted
-> ipv6 helper dhcp-snooping linkagg 2 block
-> ipv6 helper dhcp-snooping linkagg 3 client-only-trusted
```

Configuring the DHCPv6 Snooping Binding Table

To enable or disable the DHCPv6 Snooping binding table, use the **ipv6 helper dhcp-snooping binding** command. For example:

```
-> ipv6 helper dhcp-snooping binding enable
-> ipv6 helper dhcp-snooping binding disable
```

Configuring the Binding Table Timeout

The contents of the DHCPv6 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 **dhcpx6bind.db** file located in the **/flash/switch** directory.

Note. Do not manually change the **dhcpx6bind.db** file. This file is used by DHCPv6 Snooping to preserve and maintain binding table entries. Changing the file name or contents can cause problems with this functionality or with the DHCPv6 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 **ipv6 helper dhcp-snooping binding timeout** command. For example, the following command sets the timeout value to 1500 seconds:

```
-> ipv6 helper dhcp-snooping binding timeout 1500
```

Each time an automatic save is performed, the **dhcpx6bind.db** file is time stamped.

Synchronizing the Binding Table

To synchronize the contents of the **dhcpx6bind.db** file with the binding table contents that resides in memory, use the **ipv6 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 **dhcpx6bind.db** file. For example:

```
-> ipv6 helper dhcp-snooping binding action purge
-> ipv6 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 **dhcpx6bind.db** file contents into memory. This is the reverse of saving the binding table contents in memory to the **dhcpx6bind.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 32-36](#) for more information.

Binding Table Retention

When the binding table is synchronized with the contents of the **dhcpv6bind.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 **ipv6 helper dhcp-snooping binding persistency** command. For example:

```
-> ipv6 helper dhcp-snooping binding persistency enable
```

When binding table retention is enabled, entries remain in the table for the term of their DHCPv6 lease and are not removed even when the MAC address for the entry is cleared from the MAC address table.

However when the DHCP client host is connected to another port, the associated binding table is moved from the previous port to the new port.

To disable binding table retention, use the following command:

```
-> ipv6 helper dhcp-snooping binding persistency disable
```

Use the **show ipv6 helper** command to determine the status of binding table retention.

Configuring IPv6 Source Filtering

IPv6 source filtering applies to DHCPv6 Snooping ports, link aggregates, and VLANs and restricts port traffic to only packets that contain the client source MAC address, IPv6 address, and VLAN combination. The DHCPv6 Snooping binding table is used to verify the client information for the port/VLAN that is enabled for IPv6 source filtering.

Note.

- In case of OmniSwitch 6350, source filtering for both IPv4 and IPv6 cannot be configured on the same switch.
 - DHCPv6 snooping must be enabled for IPv6 source filtering to be enabled.
-

Configuring Port IPv6 Source Filtering

Port source filtering is based on the interface number, source MAC address, and source IPv6 address.

By default, IPv6 source filtering is disabled for a DHCPv6 Snooping port or a link aggregate. Use the **ipv6 helper dhcp-snooping ip-source-filter** command to enable or disable this function for a specific port, range of ports, or a link aggregate. For example,

To enable source filtering on individual port 1/1, enter:

```
-> ipv6 helper dhcp-snooping ip-source-filter port 1/1 enable
```

To enable source filtering on link aggregate 2, enter:

```
-> ipv6 helper dhcp-snooping ip-source-filter linkagg 2 enable
```

To disable source filtering, enter:

```
-> ipv6 helper dhcp-snooping ip-source-filtering port 1/1 disable
```

Configuring VLAN IPv6 Source Filtering

VLAN source filtering is based on the source VLAN ID, interface number, source MAC address, and source IPv6 address.

IPv6 source filtering can be enabled at a VLAN level and the ports associated with the VLAN when DHCPv6 snooping is enabled at the system level or VLAN level.

By default, IPv6 source filtering is disabled for a DHCP Snooping VLAN.

Use the **ipv6 helper dhcp-snooping ip-source-filter** command to enable or disable this function.

For example, to enable source filtering on VLAN 10, enter:

```
-> ipv6 helper dhcp-snooping ip-source-filter vlan 10 enable
```

Verifying the DHCPv6 Relay Configuration

To display information about the DHCPv6 Relay, use the **show** commands listed below.

For more information about the resulting displays from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*. An example of the output for the **show ipv6 helper** command is also given in “Quick Steps for Setting Up DHCP Relay” on page 32-7

| | |
|--|---|
| show ipv6 helper | Displays the current DHCPv6 Relay, Relay Agent information and DHCPv6 snooping configurations. |
| show ipv6 helper stats | Displays the IPv6 helper statistics information. |
| show ipv6 helper dhcp-snooping vlan | Displays a list of VLANs that have DHCPv6 Snooping enabled. |
| show ipv6 helper dhcp-snooping port | Displays the trust mode and DHCPv6 Snooping violation statistics for all switch ports that are filtered by DHCPv6 Snooping. |
| show ipv6 helper dhcp-snooping binding | Displays the contents of DHCPv6 Snooping binding table (database). |
| show ipv6 helper dhcp-snooping ip-source-filter | Displays the ports or VLANs on which IPv6 source filtering is enabled. |
| show ipv6 helper dhcp-snooping ip-source-filter binding | Displays the binding entries for IPv6 source filtering. |

33 Configuring DHCP Server

The Dynamic Host Configuration Protocol (DHCP) offers a framework to provide configuration information to client interfaces on a TCP/IP network. DHCP is based on the Bootstrap Protocol (BOOTP) and provides additional capabilities like dynamic allocation of reusable network addresses and configuration options.

A DHCP server provides dynamic IP addresses on lease for client interfaces on a network. It manages a pool of IP addresses and information about client configuration parameters. The DHCP server obtains an IP address request from the client interfaces. After obtaining the requests, the DHCP server assigns an IP address, a lease period, and other IP configuration parameters, such as the subnet mask and the default gateway.

This chapter describes how to configure the internal DHCP server on the OmniSwitch.

In This Chapter

This chapter describes configuration of the DHCP server and how to modify the configuration through the Command Line Interface (CLI). CLI commands are used in the configuration examples; for more details on the syntax of commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- [“DHCP Server Specifications” on page -2.](#)
- [“DHCP Server Default Values” on page -2.](#)
- [“Quick Steps to Configure Internal DHCP Server” on page -3.](#)
- [“DHCP Server Overview” on page -5](#)
- [“Interaction With Other Features” on page -6](#)
- [“Configuring DHCP Server on OmniSwitch” on page -7](#)
- [“DHCP Server Application Example” on page -10](#)
- [“Configuration File Parameters and Syntax” on page -13](#)
- [“Policy File Parameters and Syntax” on page -26](#)

DHCP Server Specifications

| | |
|---|--|
| RFCs Supported | RFC 2131 - Dynamic Host Configuration Protocol RFC 950 - Internet Standard Subnetting Procedure RFC 868 - Time Protocol RFC 1035 - Domain Implementation and Specification RFC 1191- Path MTU Discovery |
| Platforms Supported | OmniSwitch 6350, 6450 |
| DHCP Server Implementation | BOOTP/DHCP |
| UDP Port Numbers | 67 for Request and Response |
| IP address lease allocation mechanisms: | <p>Static BOOTP: IP address is allocated using the BootP configuration when the MAC address of the client is defined.</p> <p>Static DHCP: The network administrator assigns an IP address to the client. DHCP conveys the address assigned by the DHCP server to the client.</p> <p>Dynamic DHCP: The DHCP server assigns an IP address to a client for a limited period of time or until the client explicitly releases the address.</p> |
| Maximum number of leases | 2048 |
| Maximum lease information file size | 375 KB |
| DHCP server packets processing | ~50 packets per second |

DHCP Server Default Values

| Parameter Description | Command | Default Value/Comments |
|-----------------------|---------------------------------|------------------------|
| DHCP Server operation | <code>dhcp-server status</code> | disabled |

Quick Steps to Configure Internal DHCP Server

DHCP server software is installed on the OmniSwitch to centrally manage IP addresses and other TCP/IP configuration settings for clients present on a network.

Follow the steps in this section for a quick tutorial on how to configure an internal DHCP server on the OmniSwitch.

Note. For detailed information on how to configure the DHCP server on OmniSwitch, see the [Configuring DHCP Server on OmniSwitch](#) section.

1 Navigate to **/flash/switch** directory.

```
-> cd /flash/switch
```

2 Copy the **dhcpd.conf.template** file and save it as **dhcpd.conf**. The **dhcpd.conf** file can then be customized as necessary.

```
-> cp dhcpd.conf.template dhcpd.conf
```

3 Copy the **dhcpd.pcy.template** file and save it as **dhcpd.pcy**. The **dhcpd.pcy** file can then be customized as necessary.

4 Customize the **dhcpd.conf** and **dhcpd.pcy** files according to your requirements. Use the **vi** command to modify the existing configuration file.

```
-> vi dhcpd.conf
```

Declare dynamic DHCP options, global options, and server configuration parameters for client interfaces in the **dhcpd.conf** file. Add DHCP related information for a particular subnet.

For example, for the subnet 200.0.0.0, define the dynamic DHCP range, router option, domain name and other details using the following code:

```
server-identifier sample.example.com;

subnet 200.0.0.0 netmask 255.255.255.0
{
    dynamic-dhcp range 200.0.0.10 200.0.0.11
    {
        option subnet-mask 255.255.255.0;
        option routers 200.0.0.254;
        option domain-name-servers 200.0.0.99;
        option domain-name "example.com";
        option dhcp-lease-time 30000;
    }
}
```

Note. See “[Configuration File Parameters and Syntax](#)” on page -13 topic of the *Configuring DHCP Server* section for details on what each of the optional keywords specify.

5 After entering the required information in the **dhcpd.conf** file. Type **:wq** to save the changes made to the **dhcpd.conf** file.

Note.

- If the **dhcpd.conf** file is corrupted, the **dhcpd.conf.lastgood** file is used as a backup file.
 - If the **dhcpd.conf** file is updated successfully, then the **dhcpd.conf.lastgood** file is over written with the configurations present in the **dhcpd.conf** file.
 - Properly configured **dhcpd.conf** and **dhcpd.pcy** files can be transferred to the switch remotely instead of using the vi editor.
-

6 Restart the DHCP server using the **dhcp-server restart** command. The changes made in the **dhcpd.conf** file are applied to the OmniSwitch.

```
-> dhcp-server restart
```

Note. The **dhcp-server restart** command automatically updates the **dhcpd.conf**, **dhcpd.conf.lastgood** and **dhcpd.pcy** files.

7 Enable the DHCP server using the **dhcp-server** command.

```
-> dhcp-server enable
```

8 Check the IP address leases by entering the following command:

```
-> show dhcp-server leases
```

| IP address | MAC address | Lease Granted | Lease Expiry | Type |
|---------------|-------------------|---------------------|---------------------|---------|
| 200.255.91.53 | 10:fe:a2:e4:32:08 | 2010-01-16 11:38:47 | 2010-01-17 11:38:47 | Dynamic |
| 200.255.91.5 | 20:fe:a2:e4:32:08 | 2010-01-16 10:30:00 | 2010-01-18 10:30:00 | Static |
| 200.255.91.56 | 20:fe:a2:e4:33:08 | 2010-01-16 10:30:00 | 2010-01-18 10:30:00 | Dynamic |
| 200.255.91.58 | 20:fe:a2:e4:34:08 | 2010-01-16 10:30:00 | 2010-01-18 10:30:00 | Dynamic |

DHCP Server Overview

DHCP consists of two components:

- A protocol to supply client-specific configuration parameters from a DHCP server to a client.
- A mechanism to allocate network addresses to clients.

A DHCP server uses the Dynamic Host Configuration Protocol to provide initialization parameters to the clients in the network.

The DHCP process

DHCP is built on a client-server model, where a designated DHCP server allocates network addresses and delivers configuration parameters to dynamically configured client addresses. The process for a client to obtain its IP address through a DHCP server is as follows:

- 1** The client generates a DHCP request message via UDP broadcast.
- 2** The server listens for this request message.
- 3** The server responds with a DHCP reply and a valid IP address.
- 4** The server responds with a dynamic address in a defined range or one based on a MAC address.
- 5** The server leases the address for a specific time period.

Internal DHCP Server on OmniSwitch

The OmniSwitch internal DHCP server provides the abilities to:

- Enable or disable the DHCP server.
- Dynamically modify the DHCP configuration, using the `vi` editor, or through an accurately configured text file transferred to the switch.
- Restart the DHCP server.
- View the DHCP leases offered by the internal DHCP server.
- View the DHCP server statistics through the command line interface.

Interaction With Other Features

This section contains important information about the internal DHCP server and its 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.

Virtual Router Forwarding (VRF)

Address granting policies like DHCP are restricted to operate on addresses reachable through interfaces defined within the same VRF. DHCP server is supported only on the default VRF.

BootP/UDP Relay

The BootP/UDP relay is automatically disabled on the default VRF when the internal DHCP server is enabled on the switch.

DHCP Snooping

Internal DHCP server and DHCP snooping are mutually exclusive and cannot function together in the default VRF. DHCP snooping security is disabled when the DHCP server feature is enabled on the switch since the DHCP server is internal and secure.

IP Interfaces

The DHCP client gets a lease only if the switch has an IP interface and the DHCP server is configured for that particular subnet. If there are no IP address ranges defined for the incoming client interface, then the client is not assigned a lease.

In case of IP multinetting, the primary interface address is used to calculate the subnet of the interface. If there are no IP interfaces configured in the system, then the packet sent from the client is dropped.

Configuring DHCP Server on OmniSwitch

The DHCP server implementation on OmniSwitch makes use of the policy, configuration, and server database files stored in the **/flash/switch** directory. The functions of the DHCP server related files are as follows:

- **DHCP Template files:** The **dhcpd.conf.template** and **dhcpd.pcy.template** files contain the default configuration parameters and policy parameters respectively.
- **DHCP Policy file:** The **dhcpd.pcy** file initializes the global attributes for the DHCP server.
- **DHCP Configuration files:** The **dhcpd.conf** file is used to configure specific DHCP server settings on the switch such as IP address ranges and options. The **dhcpd.conf.lastgood** file is a backup for the **dhcpd.conf** file.
- **DHCP Server Database file:** The **dhcpSrv.db** file is activated only during takeover and server restart of the DHCP server. It contains lease details of the client IP addresses.

DHCP Template files

The **dhcpd.conf.template** and **dhcpd.pcy.template** files are provided as part of the AOS package. The template files are present in the **flash/switch** directory.

The **dhcpd.conf.template** (configuration template) file contains the default configuration parameters required to setup the internal DHCP server. The **dhcpd.conf.template** file provides a basic template to create a configuration file. Create a copy of the configuration template file and save it as **dhcpd.conf** in the **flash/switch** directory. Modify the **dhcpd.conf** file according to the network requirements.

The **dhcpd.pcy.template** (policy template) file contains the default policy parameters for the internal DHCP server. The **dhcpd.pcy.template** file provides a basic template to create a policy file. Create a copy of the policy template file and save it as **dhcpd.pcy** file in the **flash/switch** directory. Modify the **dhcpd.pcy** file according to the network requirements.

Policy file

The policy file is used to configure the DHCP related policies according to user requirements. The **dhcpd.pcy.template** file provides a basic template to create a policy file. The DHCP server policy parameters can be defined using the policy file. Ideally, most of the server parameters are kept static.

Example of a *dhcpd.pcy* File

```
PingDelay = 200
PingAttempts = 3
PingSendDelay = 1000
DefaultLease = 86400
```

The updated **dhcpd.pcy** file is effective only after the **dhcp-server** command on page 20-73 is **ip helper address** performed.

See the [Policy File Parameters and Syntax](#) table for additional information on individual policy parameters and how to apply the policies for internal DHCP server on the OmniSwitch.

DHCP Configuration Files

The configuration files store the network information for the DHCP clients. There are two main DHCP configuration files that can be used to configure the DHCP server on OmniSwitch. They are:

- **dhcpd.conf** file
- **dhcpd.conf.lastgood** file

The following sections provide detailed information on the **dhcpd.conf** and **dhcpd.conf.lastgood** files.

dhcpd.conf File

The **dhcpd.conf** file is used to declare DHCP options and global options for the DHCP clients. The **dhcpd.conf.template** file contains the default configuration parameters to setup the internal DHCP server. The template file can be copied to the **dhcpd.conf** file for editing.

The **dhcpd.conf** can be used to define the following:

- IP subnets
- Dynamic scopes and static bindings
- Subnet masks, DNS and default routers, and lease times
- User class or vendor class configurations

There are three types of statements in the configuration file:

- **Parameters:** Declare how, when, or what to provide to a client.
- **Declarations:** Describe the topology of the network and provide addresses for the clients. Parameters can be listed under declarations that override the global parameters.
- **Comments:** Provide a description for the parameters and declarations. Lines beginning with a hash mark (#) are considered comments and they are optional.

Example dhcpd.conf File

```
#Global parameters that specify addresses and lease time.
option domain-name-servers 200.0.0.99;
option domain-name "example.com";
option dhcp-lease-time 20000;

#IP subnet
subnet 200.0.0.0 netmask 255.255.255.0
{
    #Dynamic scope and parameters that apply to this scope overriding global params.
    dynamic-dhcp range 220.0.0.100 220.0.0.130
    {
        option routers 220.0.0.254;
        option subnet-mask 255.255.255.0;
        option domain-name "scope_example.com";
        option domain-name-servers 192.168.1.1;
        option dhcp-lease-time 30000;
    }
}
```

```
#Static binding based on MAC address
manual-dhcp 00-01-02-03-04-05 220.0.0.140
{
  option subnet-mask 255.255.255.0;
}
}
```

Note. A subnet declaration must be included for every subnet in the network related to the DHCP server.

Details about valid parameters and declarations are listed in the table - [Configuration File Parameters and Syntax](#)

dhcpd.conf.lastgood File

The **dhcpd.conf.lastgood** file is used as a backup file when the **dhcpd.conf** file is corrupted. If the **dhcpd.conf** file is affected, then the DHCP server generates an error. In such an instance, the DHCP server configuration is updated according to the **dhcpd.conf.lastgood** file. The **dhcpd.conf.lastgood** file is now used to configure the internal DHCP server, provide IP addresses on lease, and maintain DHCP related information.

The **dhcpd.conf.lastgood** file is overwritten with the configurations in the **dhcpd.conf** file when the DHCP configurations are setup or updated and the internal DHCP server is restarted successfully. At this point, the **dhcpd.conf** and **dhcpd.conf.lastgood** files are identical.

If any modifications are made in the **dhcpd.conf** file, the DHCP server must be restarted so that the configuration is updated on the OmniSwitch. The **dhcp-server** command automatically updates the **dhcpd.conf** and **dhcpd.conf.lastgood** files.

DHCP Server Database file

The **dhcpSrv.db** or the DHCP server database or lease file is initialized when the DHCP server function takes over or is restarted. The DHCP server database file contains the mappings between a client IP address and MAC address, referred to as a binding.

There are two types of bindings:

Static bindings - Map a single MAC address to a single IP address.

Dynamic bindings - Dynamically map a MAC address to an IP address from a pool of IP addresses. Details of both the dynamic and static bindings, are stored in the **dhcpSrv.db** file.

The **dhcpSrv.db** file is read when the switch reloads or the DHCP service restarts. The server database file is read-only and must not be opened or edited by the user. This file provides an account of all the subnets configured and helps in detecting all the unmanaged leases. The lease file is synchronized with the DHCP server periodically based on a timer for smooth operation during takeover and restart. The default value of this timer is 1 minute. The timer ping mechanism is used to prevent duplicate IP address allocations to clients in the same subnet. The lease file synchronization is applicable for both chassis and stack based OmniSwitch products.

DHCP Server Application Example

In this application example the clients or hosts obtain their IP addresses from the internal DHCP server configured on the OmniSwitch. DHCP clients initially have no IP address and are provided IP addresses by the DHCP server.

The external router supports the DHCP relay functionality so that it can forward DHCP frames sent to and from the DHCP clients and server on the OmniSwitch.

In the following diagram, the OmniSwitch is acting as a DHCP server and the external router is acting as the DHCP relay agent. The DHCP requests from the clients (eg: 200.0.0.X) are relayed from the external router to the OmniSwitch acting as a DHCP server. The internal DHCP server on OmniSwitch processes the requests and leases IP addresses based on the DHCP server configuration.

- 1 The DHCP clients are present in the 200.0.0.X network connected to the external router and also in the 220.0.0.X network directly attached to the OmniSwitch.
- 2 The default **dhcpd.pcy** file can be used to configure the DHCP server global parameters.
- 3 The **dhcpd.conf** file defines the 200.0.0.X network and 220.0.0.X network.
- 4 The subnet mask and DNS server address are global declarations since they are the same for each subnet.
- 5 The default router address and lease times are declared as a part of the scope since they are different for each subnet.
- 6 The resulting sample code for the **dhcpd.conf** file is as follows:

```
#Global parameters
option subnet-mask 255.255.255.0;
option domain-name-servers 200.0.0.99;
subnet 200.0.0.0 netmask 255.255.255.0
{
    dynamic-dhcp range 200.0.0.11 200.0.0.20
    {
        option routers 200.0.0.254;
        option dhcp-lease-time 20000;
    }
}

subnet 220.0.0.0 netmask 255.255.255.0
{
    dynamic-dhcp range 220.0.0.100 220.0.0.105
    {
        option routers 220.0.0.254;
        option dhcp-lease-time 30000;
    }
}
```

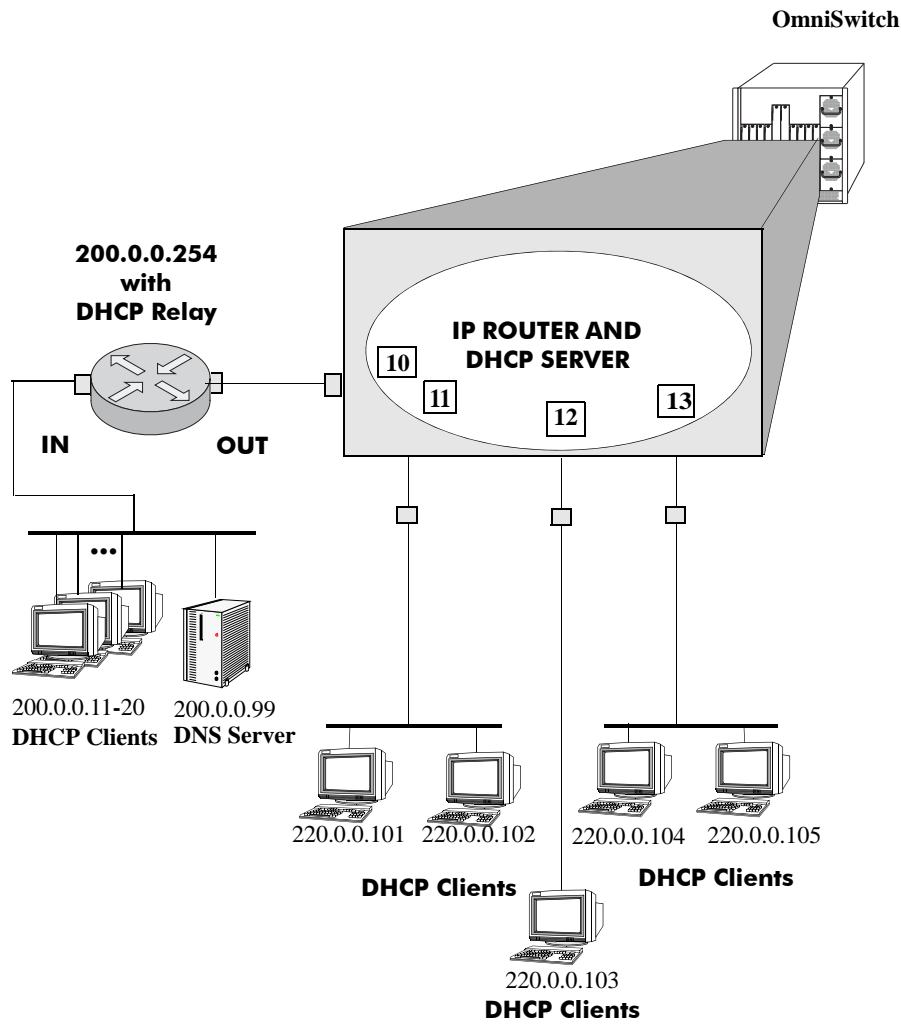


Illustration of Internal DHCP Server application example

Verifying DHCP Server Configuration

To display information about the DHCP Server configuration and statistics use the show commands listed below:

For more information about the resulting displays from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

| | |
|------------------------------------|---|
| show dhcp-server leases | Displays the leases offered by the DHCP server. |
| show dhcp-server statistics | Displays the statistics of the DHCP server. |

Configuration File Parameters and Syntax

The following table provides detailed information about the configuration file options and syntax specifications.

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|---------------------|---|-------------------------|---------------------------|--|
| 1 | subnet-mask | option subnet-mask 255.255.0.0; | N/A | Same as in Subnet Profile | Specifies the client's subnet mask. If both the subnet mask and the router option are specified in a DHCP reply, the subnet mask option must be specified before the router option. |
| 2 | time-offset | option time-offset 1000; | numeric_ signed | N/A | Specifies the offset of the client's subnet (in seconds) from Coordinated Universal Time (also referred to as UTC). A positive offset indicates a location east of the zero meridian and a negative offset indicates allocation west of the zero meridian. For example, to enter a time offset for a client subnet located in the Eastern Standard Timezone (5 hours west of the UTC zero meridian), enter -18000. |
| 3 | routers | option routers 100.0.0.1; | N/A | Same as in Subnet Profile | Lists the IP addresses for the routers for each client subnet defined. Routers should be listed in order of preference |
| 4 | time-server | option time-server 10.10.0.10; | N/A | Same as in Subnet Profile | Specifies IP address of the RFC 868 time server available to the client. |
| 5 | name-servers | option name-servers 10.10.0.100; | ip_ address_ list | N/A | Specifies IP address of the IEN-116 name server available to the client. |
| 6 | domain-name-servers | option domain-name-servers 10.10.0.30; | N/A | Same as in Subnet Profile | Lists the DNS (STD 13, RFC 1035) name server IP address(es) available to the client. Servers should be listed in order of preference. |
| 7 | log-servers | option log-servers 10.10.0.100; | ip_ address_ list | N/A | Specifies the IP address of the MIT-LCS UDP log server available to the client. |
| 8 | cookie-servers | option cookie-servers 10.10.0.100; | ip_ address_ list | N/A | Specifies the IP address of the RFC 865 cookie server available to the client. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|---------------------------|--|-----------------|---------------------------|---|
| 9 | lpr-servers | option lpr-servers 10.10.0.100; | ip_address_list | N/A | Specifies IP address of the line printer server available to the client. |
| 10 | impress-servers | option impress-servers 10.10.0.100; | ip_address_list | N/A | Specifies IP address of the Imagen Impress server available to the client. |
| 11 | resource-location-servers | option resource-location-servers 10.10.0.100; | ip_address_list | N/A | Specifies the IP address of the Resource Location server available to the client. |
| 12 | host-name | option host-name "bgp000014bgs"; | N/A | Same as in Object Profile | Specifies the name of the client. If the host name is defined in an option template, it overrides any definition in the Object Profile. |
| 13 | boot-size | option boot-size 30; | numeric | N/A | Specifies the length of the default boot image of the client. The maximum file length is 65,535 bytes. |
| 14 | merit-dump | option merit-dump "m_dump"; | text | N/A | Specifies the pathname of the file where the core image is to be dumped in the occurrence of a crash. The path is formatted as a character string consisting of characters from the Network Virtual Terminal (NVT) ASCII character set. |
| 15 | domain-name | option domain-name "abc.example.com"; | N/A | Same as in Subnet Profile | Specifies the domain name to resolve hostnames via the Domain Name Service (DNS). |
| 16 | swap-server | option swap-server 10.10.0.100; | ip address | N/A | Specifies the IP address of the client's swap server. |
| 17 | root-path | option root-path "/root"; | text | N/A | Specifies the pathname that contains the client's root directory or partition. The path is formatted as an NVT ASCII character string. |
| 18 | extensions-path | option extensions-path "/ext"; | text | N/A | Specifies a text string to denote a file, retrievable via Trivial File Transfer Protocol (TFTP). The file contains information that can be interpreted in the same way as the 64-octet vendor-extension field within the BOOTP response. The length of the file is unconstrained. All references to instances of the BOOTP Extensions Path field within the file are ignored. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|--------------------------|---|----------------------|---------------|--|
| 19 | ip-forwarding | option ip-forwarding false; | boolean | False | Select True to configure the IP layer to enable packet forwarding. Select False to disable packet forwarding. |
| 20 | non-local-source-routing | option non-local-source-routing false; | boolean | False | Select True to configure the IP layer to forward datagrams with non-local source routes. Select False to disable forwarding of the datagrams. |
| 21 | policy-filter | option policy-filter 10.10.0.100 255.255.0.0; | ip_address_mask_list | N/A | Specifies policy filters for nonlocal source routing. The filters consist of the IP address list and masks. This data specifies destination and mask pairs with which to filter incoming source routes. The client should discard any source-routed datagram whose next hop address does not match one of the filters. |
| 22 | max-dgram-reassembly | option max-dgram-reassembly 576; | numeric | N/A | Specifies the maximum reassembly size of the datagram. Enter a value between 576 and 65,535. |
| 23 | default-ip-ttl | option default-ip-ttl 1; | numeric | N/A | Specifies the default time-to-live (in seconds) to use on outgoing datagrams as an octet between 1 and 255. |
| 24 | path-mtu-aging-timeout | option path-mtu-aging-timeout 10; | numeric | N/A | Specifies the maximum time to be allotted for Path Maximum Transmit Unit (MTU) values to be discovered. The timeout is in seconds, from 0 to 2,147,483,647. |
| 25 | path-mtu-plateau-table | option path-mtu-plateau-table 68; | numeric_list | N/A | Identifies a table of MTU sizes to use when performing Path MTU discovery as defined in RFC 1191. The table is formatted as a list. Minimum value is 68. Maximum value is 65,535. |
| 26 | interface-mtu | option interface-mtu 68; | numeric | N/A | Specifies the Maximum Transmit Unit (MTU) to be used on the related interface. MTU is the frame size in a TCP/IP network. Valid range from 68 to 65,535. |
| 27 | all-subnets-local | option all-subnets-local false; | boolean | False | True indicates that all subnets share the same MTU as of the subnet to which the client user is directly connected False indicates that some of the subnets connected may have smaller MTUs. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|-----------------------------|---|----------------------|---------------------------|--|
| 28 | broadcast-address | option broadcast-address 10.10.255.255 | N/A | Same as in Subnet Profile | Specifies the broadcast address used on the client's subnet. |
| 29 | perform-mask-discovery | option perform-mask-discovery false; | boolean | False | True indicates that the client should perform subnet mask discovery. False indicates that no mask discovery must be performed. |
| 30 | mask-supplier | option mask-supplier false; | boolean | False | True indicates that response to the subnet mask request should use Internet Control Message Protocol (ICMP). False indicates the subnet mask should not respond using ICMP. |
| 31 | router-discovery | option router-discovery false; | boolean | False | True allows router discovery to be performed as defined in RFC 1256. False indicates that router discovery need not be performed. |
| 32 | router-solicitation-address | option router-solicitation-address 10.10.0.100; | ip_address | N/A | Specifies the IP address where router solicitation requests should be transmitted. |
| 33 | static-routes | option static-routes 10.10.0.100 10.10.0.200; | ip_address_pair_list | N/A | Specifies the list of static routes that the client should install in its routing cache. If multiple routes to the same destination are specified, they are listed in descending order of priority. The routes consist of a list of IP address pairs. The first address is the router for the destination. The default route (0.0.0.0) is an illegal destination for a static route. |
| 34 | "trailer-encapsulation" | option trailer-encapsulation false; | boolean | False | Select True to identify whether the client should negotiate the use of trailers (RFC 893) when using the Address Resolution Protocol (ARP). Select False to prevent the use of trailers. |
| 35 | arp-cache-timeout | option arp-cache-timeout 10; | numeric | N/A | Specifies the time-out in seconds for ARP cache entries, from 0 to 2,147,483,647. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|-------------------------|---------------------------------------|-----------------|---------------------------|--|
| 36 | ieee802-3-encapsulation | option ieee802-3-encapsulation false; | boolean | False | Use this option to identify the use of Ethernet Version 2 (RFC 894) or IEEE 802.3 (RFC 1042) encapsulation for Ethernet interface. Select True to use RFC 1042 encapsulation. Select False to use RFC 894 encapsulation. |
| 37 | default-tcp-ttl | option default-tcp-ttl 1; | numeric | N/A | Defines the default time-to-live (in seconds) to use when sending TCP segments. Enter a value from 1 to 255. |
| 38 | tcp-keepalive-interval | option tcp-keepalive-interval 10; | numeric | N/A | Specifies the amount of time, in seconds, to wait before sending a keep alive message on a TCP connection. A value of 0 indicates keep alive messages on connections should not be generated unless specifically requested to do so by an application. Valid range from 0 to 2,147,483,647 |
| 39 | tcp-keepalive-garbage | option tcp-keepalive-garbage false; | boolean | False | Specifies if the TCP keep alive messages should be sent with a garbage octet for compatibility with older implementations. Select True to enable a garbage octet to be sent. Select False to prevent a garbage octet being sent. |
| 40 | nis-domain | option nis-domain "abc.example.com"; | text | Same as in Subnet Profile | Network Information Service (NIS) support is provided on SunOS 4.1x, Solaris 2.x and HP_UX10 only. Specify the NIS domain name. The domain is formatted as a character string from the NVT ASCII character set. |
| 41 | nis-servers | option nis-servers 10.10.0.30; | ip_address_list | Same as in Subnet Profile | Lists the IP addresses (in order of preference) identifying the NIS (Network Information Service) servers available to the client |
| 42 | ntp-servers | option ntp-servers 10.10.0.50 | ip_address_list | Same as in Subnet Profile | Lists the IP addresses (in order of preference) indicating NTP (RFC 868) servers available to the client. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|----------------------|---|------------------|---------------|---|
| 43 | vendor-specific | option vendor-specific vspInfo; | hexadecimal_text | N/A | Used by clients and servers to exchange vendor-specific information. The value for this option is defined in the hexadecimal format. The definition of this information is vendor specific. The vendor is indicated in the vendor class identifier option. Servers not equipped to interpret the vendor specific information sent by a client must ignore the related data. Clients that do not receive desired vendor-specific information should attempt to operate without the related data. The clients must announce that they are working in a degraded mode. |
| 44 | netbios-name-servers | option netbios-name-servers 10.10.0.100; | ip_address_list | N/A | Specifies a list of RFC 1001/1002 NBNS name servers listed in order of preference. This is a NetBIOS name server (NBNS) or WINS server option. |
| 45 | netbios-dd-servers | option netbios-dd-servers 10.10.0.100; | ip_address_list | N/A | Specifies a list of RFC 1001/1002 NBDD servers listed in order of preference. This is a NetBIOS datagram distribution server (NBDD) option. |
| 46 | netbios-node-type | option netbios-node-type 1; | | N/A | Allows NetBIOS over TCP/IP clients, which are configurable as described in RFC 1001/1002. The value is specified as a single octet, which identifies the client type, as follows:- ValueNode type 0x1B-node 0x2P-node 0x4M-node 0x8H-node |
| 47 | netbios-scope | option netbios-scope "xyz"; | text | N/A | This NetBIOS scope option specifies the NetBIOS over TCP/IP scope parameter for the client, as specified in RFC 1001/1002. |
| 48 | font-servers | option font-servers 10.10.0.100; | ip_address_list | N/A | Specifies a list of X Window System Font servers available to the client. Servers should be listed in order of preference. |
| 49 | x-display-manager | option x-display-manager 10.10.0.100; | ip_address_list | N/A | Specifies a IP address list of systems that are running the X Window System Display Manager and are available to the client. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|----------------------|---------------------------------------|-------------------|---------------|---|
| 51 | dhcp-lease-time | option dhcp-lease-time 4294967295; | time_ interval | Unlimited | Used in a client request (DHCPDISCOVER or DHCPREQUEST) to allow the client to request a lease time for the IP address. In a server reply (DHCPOFFER), a DHCP server uses this option to specify the lease time offered. Selecting the Limited option allows you to set a lease time of up to 999 days, 999 hours, and 999 minutes. |
| 52 | dhcp-option-overload | option dhcp-option-overload 1; | 1, 2 or 3 | N/A | Used to indicate that the DHCP server name or file fields are being overloaded by using them to carry DHCP options. A DHCP server inserts this option if the returned parameters exceed the usual space allotted for options. If this option is present, the client interprets the specified additional fields after it concludes the interpretation of the standard option fields. Legal values for this option are as follows: 1 - The “file” field is used to hold options 2 - The “sname” field is used to hold options 3 - Both fields are used to hold options |
| 58 | dhcp-renewal-time | option dhcp-renewal-time 10; | numeric | N/A | Specifies the time interval from address assignment until the client transitions to the renewing state. You can enter any value from 0 to 999,999,999 seconds. |
| 59 | dhcp-rebinding-time | option dhcp-rebinding-time 10; | numeric | N/A | Specifies the time interval from address assignment until the client transitions to the rebinding state. You can enter any value from 0 to 999,999,999 seconds. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|----------------------------|--|-----------------|---------------------------|---|
| 61 | dhcp-client-identifier | option dhcp-client-identifier xyz; | text | N/A | Used by DHCP clients to specify their unique identifier. DHCP servers use this value to index their database of address bindings. It is unique for all clients in an administrative domain. The client identifier consists of type-value pairs. Ex: A hardware type and hardware address. In this case, the type field should be one of the Address Resolution Protocol (ARP) hardware types defined in RFC 1700. A hardware type - 0 indicates a domain name. Vendors and system administrators are responsible for choosing the unique client-identifiers. |
| 62 | novell-netware-domain-name | option novell-netware-domain-name "xyz"; | text | N/A | Used to convey the NetWare/IP domain name used by the NetWare/IP product. The NetWare/IP Domain in the option is a Network Virtual Terminal (NVT) ASCII text string. You can enter up to 255 characters. |
| 63 | novell-netware-info | option novell-netware-info [0100]; | sub-option | N/A | This NetWare/IP option code is used to convey all the NetWare/IP related information except for the NetWare/IP domain name. If NWIP_EXIST_IN_OPTIONS _AREA sub-option is set, one or more of the other suboptions may be present. |
| 64 | dhcp-nis+-domain | option dhcp-nis+-domain "xyz"; | text | Same as in Subnet profile | Specifies the NIS domain name. The domain is formatted as a character string from the NVT ASCII character set Network Information Service (NIS) support is provided on SunOS 4.1x, Solaris 2.x and HP_UX10 only. |
| 65 | dhcp-nis+-servers | option dhcp-nis+-servers 10.10.0.100; | ip_address_list | Same as in Subnet profile | Lists the IP addresses identifying the NIS servers available to the client in order of preference |
| 66 | dhcp-tftp-server | option dhcp-tftp-server "xyz"; | text | N/A | Used to identify a Trivial File Transfer Protocol (TFTP) server when the server name field in the DHCP header has been used for DHCP options. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|---------------------------|--|---------------------|---------------|--|
| 67 | dhcp-bootfile-name | option dhcp-bootfile-name "xyz"; | text | N/A | This option is used to identify a bootfile when the file field in the DHCP header has been used for DHCP options. |
| 68 | dhcp-mobile-ip-home-agent | option dhcp-mobile-ip-home-agent 10.10.0.100; | ip_address_ list | N/A | This option specifies an IP address list indicating mobile IP home agents available to the client. Agents should be listed in order of preference. |
| 69 | dhcp-smtp-server | option dhcp-smtp-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of SMTP servers available to the client. Servers should be listed in order of preference. |
| 70 | dhcp-pop3-server | option dhcp-pop3-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of POP3 servers available to the client. Servers should be listed in order of preference. |
| 71 | dhcp-nntp-server | option dhcp-nntp-server 10.10.0.100; | ip_address_ list | N/A | This Network News Transport Protocol (NNTP) server option specifies a list of NNTP servers available to the client. Servers should be listed in order of preference. |
| 72 | dhcp-www-server | option dhcp-www-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of WWW servers available to the client. Servers should be listed in order of preference. |
| 73 | dhcp-finger-server | option dhcp-finger-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of Finger servers available to the client. Servers should be listed in order of preference. |
| 74 | dhcp-irc-server | option dhcp-irc-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of IRC servers available to the client. Servers should be listed in order of preference. |
| 75 | dhcp-street-talk-server | option dhcp-streetalk-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of StreetTalk servers available to the client. Servers should be listed in order of preference. |
| 76 | dhcp-stda-server | option dhcp-stda-server 10.10.0.100; | ip_address_ list | N/A | Specifies a list of STDA (Street-Talk Directory Assistance) servers available to the client. Servers should be listed in order of preference. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|------------------------------------|--|-----------------|---------------|--|
| 78 | slp-directory-agent | option slp-directory-agent [000a0a0064]; | sub-option | | Specifies the location of one or more SLP Directory Agents. The SLP Directory Agent option contains the following suboptions: |
| | Mandatory | | boolean | False | This sub-option may be set to either True or False. If it is set to True, the SLP UserAgent or Service Agent so configured must not employ either active or passive multicast discovery of Directory Agents. |
| | Directory Agent Address | | ip_address_list | N/A | This sub-option allows a IP address list to be specified. The list must be in order of preference, if an order of preference is desired. |
| 79 | slp-service-scope | option slp-service-scope [0078797a]; | sub-option | | Specifies the scopes that a SLP Agent is configured to use. It contains the following suboptions: If set to False , static configuration takes precedence over the DHCP provided scope list. If set to True , the entries in the Scope List must be used by the SLP Agent. |
| | Scope Listtext | | | N/A | This sub-option is a comma-delimited list of scopes. The list is case insensitive. |
| | Mandatory | | boolean | FALSE | This sub-option determines whether SLP Agents override their static configuration for scopes in the Scope List. This allows DHCP administrators to implement a policy of assigning a set of scopes to Agents for service provision. |
| 85 | novell-nds-servers | option novell-nds-servers 10.10.0.100; | ip_address_list | N/A | Specifies one or more NDS servers for the client to contact for access to the NDS database. Servers should be listed in order of preference. |
| 86 | novell-nds-tree-name | option novell-nds-tree-name "xyz"; | text | N/A | Specifies the name of the NDS tree which the client can contact. Maximum 255 characters. |
| 87 | novell-nds-context | option novell-nds-context "xyz"; | text | N/A | Specifies the initial NDS context the client should use. Maximum 255 characters. |
| 88 | broadcast-multicast-service-domain | option broadcast-multicast-service-domain [0378797a00]; | name_list | N/A | Lists server names that host the Broadcast and Multicast services that are specified as domain names. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|-------------------------------------|---|-----------------|---------------|--|
| 89 | broadcast-multicast-service-address | option broadcast-multicast-service-address 10.10.0.100; | ip_address_list | N/A | Lists server names that host the Broadcast and Multicast services that are specified as IPV4 addresses. |
| 98 | user-authentication-protocol | option user-authentication-protocol [78797a]; | text_list | N/A | Specifies a list of Uniform Resource Locators (URLs), each pointing to a user authentication service that is capable of processing authentication requests encapsulated in the UAP. UAP servers can accept either HTTP 1.1 or SSLv3 connections. If the list includes a URL that does not contain a port component, the normal default port is assumed (port 80 for http and port 443 for https). If the list includes a URL that does not contain a path component, the path /uap is assumed. |
| 100 | timezone-posix | option timezone-posix "xyz"; | text | 255 | Specifies a DHCP client's timezone specified as a POSIX 1003.1 timezone string. |
| 101 | timezone-database | option timezone-database "xyz"; | text | 255 | Specifies a DHCP client's timezone specified as a TZ database string. |
| 116 | ipv4-auto-configuration | option ipv4-auto-configuration false; | boolean | False | This option is used to check whether, and be notified if, auto-configuration should be disabled on the local subnet. When a server responds with the value "AutoConfigure" (True), the client may generate a linklocal IP address if appropriate. However, if the server responds with "DoNotAutoConfigure" (False), the client must not generate a link-local IP address, possibly leaving it with no IP address. |
| 119 | domain-search | option domain-search [0378797a00]; | text_list | N/A | Passes the domains in the search list from the DHCP Server to the DHCP Client to use when resolving hostnames using DNS. |
| 120 | sip-server | option sip-server [010a0a0064]; | ip_address_list | N/A | Lists the SIP servers specified as IPV4 |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|------------------------------------|--|-----------------|---------------|--|
| 121 | classless-static-route | option classless-static-route 16.10.10.10.10.0.200; | ip_mask_ip_list | N/A | Specifies one or more static routes, each of which consists of a destination descriptor (the subnet address and subnet mask) and the IP address of the router that should be used to reach that destination. |
| 122 | cablelabs-client-config | option cablelabs-client-config [01040a0a00640 2040a0a0064040 c0000000a00000 00a0000000a050 c0000000a00000 00a0000000a060 50358595a00070 100080100]; option 177 [010378797a020 378797a0303787 97a040378797a0 50378797a06037 8797a070100080 100090378797a] ; | sub_option | | The following table describes the CableLabs Client Configuration 122 sub-options, specified in RFC 3495: |
| | TSP Primary DHCP Server Address | | ip_address | N/A | Specifies the IP address of the TSP's primary DHCP server from which an MTA is permitted to accept a DHCP OFFER. |
| | TSP Secondary DHCP Server Address | | ip_address | N/A | Specifies the IP address of the TSP's secondary DHCP server from which an MTA is permitted to accept a DHCP OFFER. |
| | TSP Provisioning Server Address | | | IP_address | MTAs communicate with the Provisioning server at various stages in their provisioning process. Enter either the IP address or the FQDN of the TSP's Provisioning server. |
| | TSP ASREQ/AS-REP Backoff and Retry | | sub-option | N/A | Configures an MTA's Kerberos ASREQ/AS-REP timeout, backoff, and retry mechanism. Enter a Nominal Timeout value in milliseconds, a Maximum Timeout value in seconds and a Maximum Retry value. All these values are unsigned. |

| Option Code | dhcpd.conf Key-word | dhcpd.conf example | Data type | Default Value | Description |
|-------------|--|--------------------|-----------|---------------|--|
| | TSP Kerberos Realm Name | | text | N/A | The Packet Cable architecture requires an MTA to authenticate itself to the TSP's network through the Kerberos protocol. A Kerberos Realm name is required at the MTA to permit a DNS lookup for the address of the TSP's Kerberos Key Distribution Center (KDC) entity. Note: The realm name must be all capital letters and conform to domain name syntax (HOST.SUB-DOMAIN.DOMAIN). |
| | TSP Ticket Granting Server Utilization | | boolean | False | Determines whether an MTA should use a Ticket Granting Ticket (TGT) when obtaining a service ticket for one of the PacketCable application servers. Select True to indicate that the MTA should get its TGT. |
| | TSP Provisioning Timer | | numeric | 0 | Defines the maximum time allowed for the MTA provisioning process to complete. If this timer expires before the MTA has completed the provisioning process, the MTA should reset the timer and re-start its provisioning process from the beginning. Enter a value from 0 to 255, where 0 means the timer is disabled. |

Policy File Parameters and Syntax

| Num | Policy | Usage | Default Value | Description |
|-----|-----------------------------|-------------------------------------|-------------------|---|
| 1 | ActiveLease Expiration | ActiveLease Expiration = On | Off | <p>Determines how the expired leases are handled. The following values are available:</p> <p>Off - prevents expired leases from being automatically deleted after lease period is over.</p> <p>Full_delete - causes the lease from DHCP database to be deleted, and the Message Service to be notified of expired leases.</p> |
| 2 | Check TransactionID | Check Transaction ID=True | False | Configures the service to ignore multiple discover, request, and BootP messages that have the same XID. |
| 3 | DefaultLease | Default Lease=86400 | 7776000 (90 days) | Specifies the default lease period provided for the clients in seconds. |
| 4 | DropAll DhcpInform Packets | DropAll Dhcp Inform Packets = True | False | <p>Allows administrators to configure the DHCP server to ignore inform packets.</p> <p>If this policy is set to True, the DHCP server prevents the processing of DHCPINFORM packets. However, the incoming packets are parsed.</p> |
| 4 | DropZero MacAddress Packets | DropZero MacAddress Packets = False | True | <p>If this policy is set to True, the DHCP server checks all incoming packets for a zero MAC address and drops the packet if it is found.</p> <p>Note: DHCPINFORM messages are processed even if this process is set to true.</p> |
| 5 | ForceClass | ForceClass =VendorNone | True | <p>Determines if the service verifies the lease request from the client before issuing a lease.</p> <p>The values associated with this policy are as follows:</p> <ul style="list-style-type: none"> • None - Allows the server to issue leases from any IP address range to an incoming client request. • Both - Forces the service to match for both user and vendor class with the values defined for a particular IP address range. • Vendor - The service must match only on the vendor class. • User - The service must match only on user class. |

| Num | Policy | Usage | Default Value | Description |
|-----|---------------------------------|---|---------------|--|
| 6 | Honor Requested LeaseTime | Honor Requested Lease Time = False | True | <p>If this policy is set to True, the DHCP server provides the requested lease time to the client.</p> <p>If this policy is set to False, the server offers the configured lease time.</p> |
| 7 | Lease Expiration SleepTime | Lease Expiration SleepTime = 120000 | 60000 msec | <p>Specifies the time interval in milli seconds after which the lease expiration processing occurs.</p> <p>Note: This value must not be less than 1 minute.</p> |
| 8 | MaxPending Seconds | MaxPending Seconds = 20 | 10 | <p>Specifies the number of seconds that an offered lease remains in a pending state.</p> <p>When a client sends a DHCPDISCOVER request, the DHCP server responds with a DHCPOFFER and offers an IP address. The address is marked as pending for the specified period of time.</p> |
| 9 | Max Unavailable Time | Max Unavailable Time = 14000 | 86400 | Determines the period of time that an IP address is not available after a DHCPDECLINE or ping packet is sent as response. After this time period, the server considers this address as available. |
| 10 | Nak Unknown Clients | NakUnknown Clients = False | True | <p>Prevents the DHCP server from providing DHCP addresses to clients which are not in the defined subnets of the DHCP server.</p> <p>This policy must be set to False in environments where multiple DHCP services are active in the same subnet or subnets.</p> |
| 11 | NackDhcp RequestsFor Duplicates | NackDhcp RequestsFor Duplicates = False | True | <p>If this value is set to True, the DHCP server sends a NAK if a RENEW/REBIND request or SELECTING request is received for an IP address already owned by another hardware interface.</p> <p>If this value is set to False, the invalid request is dropped.</p> |
| 12 | PingAttempts | Ping Attempts = 3 | 1 | Specifies the number of attempts to ping through which DHCP server determines if the IP address is already in use. |
| 13 | PingDelay | PingDelay = 200 | N/A | Specifies the delay in milliseconds between two consecutive pings to check the IP address usage in the network. |
| 14 | PingSendDelay | PingSend Delay = 1000 | 500 | Specifies the number of milliseconds between subsequent pings. This is applicable only if the ping attempts are greater than 1. If the value of PingAttempts is greater than 1, then the PingSendDelay overrides the PingDelay policy. |

| Num | Policy | Usage | Default Value | Description |
|-----|------------------------------|-------------------------------------|---------------|--|
| 15 | PingRetention | PingRetention = 200 | 0 | Specifies the number of seconds for which a ping is valid. If a ping is attempted and no response is returned, then the address is considered to be available. During the ping retention period, other ping requests are ignored. |
| 18 | PingBeforeManualDhcp | PingBeforeManualDhcp = False | True | If this value is set to True , the DHCP server performs a ping before assigning a static DHCP address. If an ICMP_REPLY is received from the ping, then the DHCP offer is not sent to the client and the address is marked as unavailable. |
| 19 | PingBeforeManualBootp | PingBeforeManualBootp = True | False | If this value is set to True , the DHCP server performs a ping before assigning a static BootP address. If an ICMP_REPLY is received, the BootP reply is not sent to the client, and the BootP address is marked as unavailable. |
| 20 | RegisteredClientsOnly | RegisteredClientsOnly = True | False | <p>This policy is used when the MAC pool addresses are defined at either the global or the subnet level.</p> <p>If this value is set to True, the DHCP information is provided to the clients that have a known MAC address (configured in a MAC pool). If MAC pool addresses are not defined at either the global or the subnet level, the none of the devices are provided a DHCP lease.</p> <p>If this value is set to False, the DHCP information is provided to all clients.</p> |
| 21 | SendRequestedParamsOnly | SendRequestedParamsOnly = True | False | <p>If this value is set to True, the DHCP server sends only the options requested by the client. For example, if the client sends a DHCP parameter request list - option (55) in the Discover packet, then the server sends only the options that are both configured and requested by the client. The subnet-mask (1) and lease-time (51) options are always sent to the client, in addition to the IP address.</p> <p>If this value is set to False, the service sends all the configured options to the client.</p> |
| 22 | SupportRelayAgentDeviceClass | SupportRelayAgentDeviceClass = True | False | If this policy is set to True , the server supports the assignment of DHCP options by the DOCSIS device class. |
| 23 | ZeroCiAddr | ZeroCiAddr = True | False | <p>This policy affects the contents of the “ciaddr” field in outgoing packets.</p> <p>If this policy is set to True, the service fills in “ciaddr” with 0.0.0.0 on reply (ACK) packets.</p> |

34 Configuring VRRP

The Virtual Router Redundancy Protocol (VRRPv2/VRRPv3) is a standard router redundancy protocol supported in IPv4/IPv6, based on RFC 3768 and RFC 2787. It provides redundancy by eliminating the single point of failure inherent in a default route environment. The VRRPv2/VRRPv3 router, which controls the IPv4/IPv6 address associated with a virtual router is called the master router, and is responsible for forwarding virtual router advertisements. If the master router becomes unavailable, the highest priority backup router transitions to the master state. The Alcatel-Lucent implementation of VRRP also supports the collective management of virtual routers on a switch.

Note. The VRRPv3 implementation is based on the latest Internet Draft, Virtual Router Redundancy Protocol for IPv6, September 2004.

Note. RFC 3768, which obsoletes RFC 2338, does not include support for authentication types. As a result, configuring VRRP authentication is no longer supported in this release.

In This Chapter

This chapter describes VRRPv2/VRRPv3 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 AOS Release 6 CLI Reference Guide*.

This chapter provides an overview of VRRP and includes information about the following:

- Virtual routers—see [“Creating/Deleting a Virtual Router”](#) on page 34-10.
- IP addresses for virtual routers—see [“Specifying an IP Address for a Virtual Router”](#) on page 34-11.
- VRRP advertisement interval—see [“Configuring the Advertisement Interval”](#) on page 34-12.
- Virtual router priority—see [“Configuring Virtual Router Priority”](#) on page 34-12.
- Preempting virtual routers—see [“Setting Preemption for Virtual Routers”](#) on page 34-13.
- VRRP authentication—see [“Configuring VRRP Authentication”](#) on page 34-13.
- VRRP traps—see [“Setting VRRP Traps”](#) on page 34-15.
- Configuring Collective Management Functionality— see [“Configuring Collective Management Functionality”](#) on page 34-16
- Verifying the VRRP configuration—see [“Verifying the VRRP Configuration”](#) on page 34-19.
- VRRPv3 Virtual routers—see [“VRRPv3 Configuration Overview”](#) on page 34-20.

- IPv6 addresses for VRRPv3 virtual routers—see [“Specify an IPv6 Address for a VRRPv3 Virtual Router”](#) on page 34-22.
- Accept mode for master router—see [“Configuring the VRRPv3 Advertisement Interval”](#) on page 34-22.
- VRRPv3 advertisement interval—see [“Configuring the VRRPv3 Advertisement Interval”](#) on page 34-22.
- VRRPv3 Virtual router priority—see [“Configuring the VRRPv3 Virtual Router Priority”](#) on page 34-23.
- Preempting VRRPv3 virtual routers—see [“Setting Preemption for VRRPv3 Virtual Routers”](#) on page 34-23.
- VRRPv3 traps—see [“Setting VRRPv3 Traps”](#) on page 34-25.
- VRRP tracking—see [“Creating Tracking Policies”](#) on page 34-27.
- VRRPv3 tracking—see [“Creating Tracking Policies”](#) on page 34-27.
- Verifying the VRRP configuration—see [“Verifying the VRRPv3 Configuration”](#) on page 34-26.

VRRP Specifications

| | |
|--|---|
| RFCs Supported | RFC 3768–Virtual Router Redundancy Protocol RFC 2787–Definitions of Managed Objects for the Virtual Router Redundancy Protocol |
| Platforms Supported | OmniSwitch 6350, 6450 |
| Compatible with HSRP? | No |
| Maximum number of VRRPv2 and VRRPv3 virtual routers combined | 255 per switch |
| Maximum number of IP addresses | 255 per virtual router |

VRRP Defaults

The following table lists the defaults for VRRP configuration through the **vrrp** command and the relevant command keywords:

| Description | Keyword | Default |
|------------------------------------|---------------------------------------|------------------------------------|
| Virtual router enabled or disabled | enable disable on off | Virtual routers are disabled (off) |
| Priority | priority | 100 |
| Preempt mode | preempt no preempt | Preempt mode is enabled |
| Advertising interval | advertising interval | 1 second |
| VRRP authentication | authenticate no authenticate | Authentication is not enabled. |

The following table lists the defaults for VRRP configuration using the VRRP collective management features and the relevant command:

| | | |
|--|----------------------|----------------|
| Default advertising interval for all the virtual routers on the switch. | vrrp interval | 1 second |
| Default priority value for all the virtual routers on the switch. | vrrp priority | 100 |
| Default preempt mode for all the virtual routers on the switch. | vrrp preempt | preempt |
| Parameter value that is to be set and/or override with the new default value in all the virtual routers on the switch. | vrrp set | all |
| Default advertising interval for all the virtual routers in the group. | vrrp group | 1 |
| Default priority value for all the virtual routers in the group. | vrrp group | 100 |

| | | |
|--|-------------------|----------------|
| Default preempt mode for all the virtual routers in the group. | vrrp group | preempt |
|--|-------------------|----------------|

| | | |
|---|-----------------------|------------|
| Parameter value that is to be set and/or override with the new default value in all the virtual routers in the group. | vrrp group set | all |
|---|-----------------------|------------|

In addition, other defaults for VRRP include:

| Description | Command | Default |
|--------------------|-------------------|----------------|
| VRRP traps | vrrp track | Disabled |
| VRRP delay | vrrp delay | 45 seconds |

Quick Steps for Creating a Virtual Router

- 1 Create a virtual router. Specify a virtual router ID (VRID) and a VLAN ID. For example:

```
-> vrrp 6 4
```

The VLAN must already be created on the switch. For information about creating VLANs, see [Chapter 4, “Configuring VLANs.”](#)

- 2 Typically, VRRP authentication should be set up with a password. For example:

```
-> vrrp 6 4 authenticate wwwtoe
```

- 3 Configure an IP address for the virtual router.

```
-> vrrp 6 4 address 10.10.2.3
```

- 4 Repeat steps 1 through 3 on all of the physical switches that participates in backing up the addresses associated with the virtual router. The authentication password must be the same on each switch.

- 5 Enable VRRP on each switch.

```
-> vrrp 6 4 enable
```

Note. *Optional.* To verify the VRRP configuration, enter the [show vrrp](#) command. The display is similar to the one shown here:

```
VRRP trap generation: Enabled
VRRP startup delay: 45 (expired)
      IP           Admin
VRID  VLAN  Address(es)  Status      Priority  Preempt  Adv
-----+-----+-----+-----+-----+-----+-----
  6    4    10.10.2.3   Enabled      100      yes      1
```

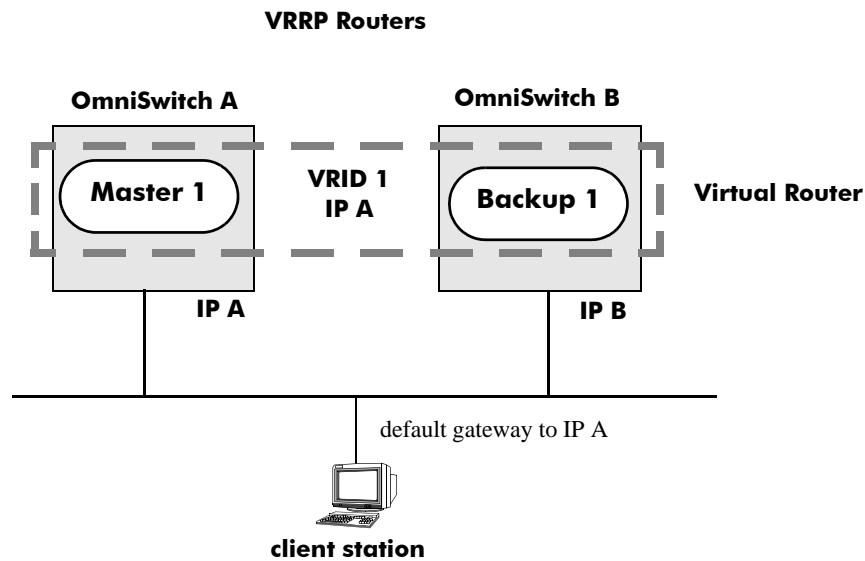
For more information about this display, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

VRRP Overview

VRRP allows the routers on a LAN to back up a default route. VRRP dynamically assigns responsibility for a virtual router to a physical router (VRRP router) on the LAN. The virtual router is associated with an IP address (or set of IP addresses) on the LAN. A virtual router master is elected to forward packets to the IP address of the virtual router. If the master router becomes unavailable, the highest priority backup router transitions to the master state.

Note. The IP address that is backed up can be the IP address of a physical router, or it can be a virtual IP address.

The example provided here is intended for understanding VRRP and does not show a configuration that would be used in an actual network.



VRRP Redundancy Example

In this example, each physical router is configured with a virtual router, VRID 1 which is associated with IP address A. OmniSwitch A is the master router because it contains the physical interface to which IP address A is assigned. OmniSwitch B is the backup router. The client is configured with a gateway address of IP A.

When VRRP is configured on these switches, and both the switches are available, OmniSwitch A responds to ARP requests for IP address A using the MAC address (00:00:5E:00:01:01) of the virtual router instead of the physical MAC address assigned to the interface. OmniSwitch A accepts the packets sent to the virtual MAC address and forward them as appropriate; it also accepts the packets addressed to IP address A (such as ICMP ping requests).

OmniSwitch B responds to ARP requests for IP address B using the physical MAC address of the interface. It does not respond to ARP requests for IP address A or to the virtual router MAC address.

If OmniSwitch A becomes unavailable, OmniSwitch B becomes the master router. OmniSwitch B then responds to ARP requests for IP address A using the MAC address (00:00:5E:00:01:01) of the virtual router. It also forwards the packets for IP address B and respond to ARP requests for IP address B using the OmniSwitch's physical MAC address.

OmniSwitch B uses IP address B to access the LAN. However, IP address B is not backed up. Therefore, when OmniSwitch B becomes unavailable, IP address B also becomes unavailable.

Why Use VRRP?

An end host can use dynamic routing or router discovery protocols to determine its first hop toward a particular IP destination. With dynamic routing, large timer values are required and can cause significant delay in the detection of a dead neighbor.

If an end host uses a static route to its default gateway, this creates a single point of failure if the route becomes unavailable. End hosts will not be able to detect alternate paths.

In either case, VRRP ensures that an alternate path is always available.

Definition of a Virtual Router

To back up an IP address or addresses using VRRP, a virtual router must be configured on VRRP routers on a common LAN. A VRRP router is a physical router running VRRP. A virtual router is defined by a virtual router identifier (VRID) and a set of associated IP addresses on the LAN.

Note. A limitation of the OmniSwitch is that a single VRID can be associated with a VLAN.

Each VRRP router can back up one or more virtual routers. The VRRP router containing the physical interfaces to which the virtual router IP addresses are assigned is called the *IP address owner*. If it is available, the IP address owner functions as the master router. The master router assumes the responsibility of forwarding packets sent to the IP addresses associated with the virtual router and answering ARP requests for these addresses.

To minimize network traffic, only the master router sends VRRP advertisements on the LAN. The IP address assigned to the physical interface on the current master router is used as the source address in VRRP advertisements. The advertisements communicate the priority and state of the master router associated with the VRID to all VRRP routers. The advertisements are IP multicast datagrams sent to the VRRP multicast address 224.0.0.18 (as determined by the Internet Assigned Numbers Authority).

If a master router becomes unavailable, it stops sending VRRP advertisements on the LAN. The backup routers know that the master is unavailable based on the following algorithm:

$$\text{Master Down Interval} = (3 * \text{Advertisement Interval}) + \text{Skew Time}$$

where *Advertisement Interval* is the time interval between VRRP advertisements, and *Skew Time* is calculated based on the priority value of the VRRP router as follows:

$$\text{Skew Time} = (256 - \text{Priority}) / 256$$

If the backup routers are configured with priority values that are close in value, there can be a timing conflict, and the first backup to take over cannot be the one with the highest priority; and a backup with a higher priority then preempts the new master. The virtual router can be configured to prohibit any

preemption attempts, except by the IP address owner. An IP address owner, if it is available, always becomes the master of any virtual router associated with its IP addresses.

Note. Duplicate IP address or MAC address messages may display when a backup router takes over for a master, depending on the timing of the takeover and the configured advertisement interval. This is true if more than one backup is configured.

VRRP MAC Addresses

Each virtual router has a single well-known MAC address, which is used as the source in all periodic VRRP advertisements sent by the master router, as the MAC address in ARP replies sent by VRRPv2, and as the MAC address in neighbor advertisements sent by VRRPv3 (instead of the MAC address for the physical VRRP router).

The VRRPv2 (IPv4) address has the following format:

00-00-5E-00-01-[virtual router ID]

The VRRPv3 (IPv6) address has the following format:

00-00-5E-00-01-[virtual router ID]

This mapping provides for up to 255 virtual routers (VRRPv2 and VRRPv3 combined) on an OmniSwitch.

ARP Requests

Each virtual router has a single well-known MAC address, which is used as the MAC address in ARP instead of a VRRP router's physical MAC address. When an end host sends an ARP request to the IP address of the master router, the master router responds to the ARP request using the virtual router MAC address. If a backup router takes over for the master, and an end host sends an ARP request, the backup replies to the request using the virtual router MAC address.

Gratuitous ARP requests for the virtual router IP address or MAC address are broadcast when the OmniSwitch becomes the master router. For VRRP interfaces, gratuitous ARP requests are delayed at system boot until both the IP address and the virtual router MAC address are configured.

If an interface IP address is shared by a virtual router, the routing mechanism does not send a gratuitous ARP for the IP address (since the virtual router sends a gratuitous ARP). This prevents traffic from being forwarded to the router before the routing tables are stabilized.

ICMP Redirects

ICMP redirects are not sent out over VRRP interfaces.

VRRP Startup Delay

When a virtual router reboots and becomes master, it can become master before its routing tables are populated. This could result in loss of connectivity to the router. To prevent the loss in connectivity, a delay is used to prevent the router from becoming master before the routing tables are stabilized; the default delay value is 45 seconds.

The startup delay can be modified to allow more or less time for the router to stabilize its routing tables.

In addition to the startup delay, the switch has an ARP delay (which is not configurable).

VRRP Tracking

Priority of the virtual router can be conditionally modified to prevent another router from taking over as master. Tracking policies are used to conditionally modify the priority setting whenever a slot/port, IP address and or IP interface associated with a virtual router goes down.

A tracking policy consists of a tracking ID, the value used to decrease the priority value, and the slot/port number, IP address, or IP interface name to be monitored by the policy. The policy is then associated with one or more virtual routers.

Configuring Collective Management Functionality

This feature provides user with the flexibility to manage the virtual routers on the switch collectively and also the capability to group the virtual routers to a virtual router group which simplifies the configuration and management tasks.

You can change the default values of the parameters like advertising interval, priority, preempt mode and the administrative status of all the virtual routers on a switch or in a virtual router group using this collective management functionality feature. For more information about configuring collective management functionality, see [page 34-16](#).

Note. VRRP3 does not support the collective management functionality in this release.

Interaction With Other Features

- IP routing—IP routing must be enabled for the VRRP configuration to take effect.
- Router Discovery Protocol (RDP)—If RDP is enabled on the switch, and VRRP is enabled, RDP advertises VLAN IP addresses of virtual routers depending on whether there are virtual routers active on the LAN, and whether those routers are backups or masters. When there are no virtual routers active on the VLAN (either acting as master or backup), RDP advertises all VLAN IP addresses. However, if virtual routers are active, RDP advertises IP addresses for any master routers; RDP does not advertise IP addresses for backup routers.

For more information about RDP, see [Chapter 31, “Configuring RDP.”](#)

VRRP Configuration Overview

During startup, VRRP is loaded onto the switch and is enabled. Virtual routers must be configured and enabled as described in the following sections. Since VRRP is implemented on multiple switches in the network, some VRRP parameters must be identical across switches:

- **VRRP and ACLs**

If QoS filtering rules (Access Control Lists) are configured for Layer 3 traffic on a VRRP router, all of the VRRP routers on the LAN must be configured with the same filtering rules; otherwise the security of the network is compromised. For more information about filtering, see

- [Chapter 40, “Configuring ACLs.”](#)

- **Conflicting VRRP Parameters Across Switches**

All virtual routers with the same VRID on the LAN must be configured with the same advertisement interval and IP addresses, and authentication password. If the virtual routers are configured differently, it can result in more than one virtual router acting as the master router. This in turn would result in duplicate IP and MAC address messages as well as multiple routers forwarding duplicate packets to the virtual router MAC address. Use the [show vrrp statistics](#) command to check for conflicting parameters. For information about configuring VRRP parameters, see the remaining sections of this chapter.

Basic Virtual Router Configuration

At least two virtual routers must be configured on the LAN—a master router and a backup router. The virtual router is identified by a number called the Virtual Router ID (VRID), the VLAN on which the virtual router is configured, and the IP address or addresses associated with the router. Multiple virtual routers can be configured on a single physical VRRP router.

Basic commands for setting up virtual routers include:

```
vrrp  
vrrp address
```

The next sections describe how to use these commands.

Creating/Deleting a Virtual Router

To create a virtual router, enter the **vrrp** command with the desired VRID and the relevant VLAN ID. The VRID must be a unique number in the range from 1 to 255. The VLAN must already be created on the switch through the **vlan** command. For information about creating VLANs, see [Chapter 4, “Configuring VLANs.”](#) For example:

```
-> vrrp 6 4
```

This command creates VRID 6 on VLAN 4.

When you create a new virtual router, the VRID ID and a VLAN ID are *required*. Optionally, you can also specify:

- **Priority** (in the range from 1 to 255); use the **priority** keyword with the desired value. The default is 100. The IP address owner is automatically assigned a value of 255, which overrides any value already configured. See [“Configuring Virtual Router Priority” on page 34-12](#) for more information about how priority is used.

- **Preempt mode.** By default, preempt mode is enabled. Use **no preempt** to turn it off, and **preempt** to turn it back on. For more information about the preempt mode, see [“Setting Preemption for Virtual Routers” on page 34-13](#).
- **Advertising interval** (in seconds). Use the **interval** keyword with the desired number of seconds for the delay in sending VRRP advertisement packets. The default is 1 second. See [“Configuring the Advertisement Interval” on page 34-12](#).
- **VRRP authentication.** By default, VRRP packets are not authenticated; use **authenticate** to enable simple text password authentication. A password of up to 16 characters must be entered. Use **no authenticate** to disable authentication. See [“Configuring VRRP Authentication” on page 34-13](#).

The following example creates a virtual router (with VRID 7) on VLAN 2 with a priority of 75. The preempt mode of the router is enabled and VRRP advertisements are sent at intervals of 2 seconds, and VRRP packets will be authenticated with the password **wwwtoe**:

```
-> vrrp 7 2 priority 75 preempt interval 2
```

Note. All virtual routers with the same VRID on the same LAN must be configured with the same advertising interval; otherwise the network can produce duplicate IP or MAC address messages. These virtual routers should be configured with the same authentication setting. If authentication is enabled, all routers must have the same password.

The **vrrp** command can also be used to specify whether the virtual router is enabled or disabled (it is disabled by default). *However, the virtual router must have an IP address assigned to it before it can be enabled.* Use the **vrrp address** command as described in the next section to specify an IP address or addresses.

To delete a virtual router, use the **no** form of the **vrrp** command with the relevant VRID and VLAN ID. For example:

```
-> no vrrp 7 3
```

Virtual router 7 on VLAN 3 is deleted from the configuration. (The virtual router does not have to be disabled before you delete it.)

For more information about the **vrrp** command syntax, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Specifying an IP Address for a Virtual Router

An IP address must be specified before a virtual router can be enabled. To specify an IP address for a virtual router, use the **vrrp address** command and the relevant IP address. For example:

```
-> vrrp 6 4 address 10.10.2.3
-> vrrp 6 4 enable
```

In this example, the **vrrp address** command specifies that virtual router 6 on VLAN 4 is used to back up IP address 10.10.2.3. The virtual router is then enabled with the **vrrp** command.

If a virtual router is to be the IP address owner, then all addresses on the virtual router must match an address on the switch interface.

To remove an IP address from a virtual router, use the **no** form of the **vrrp address** command. For example:

```
-> vrrp 6 4 disable
-> vrrp 6 4 no address 10.10.2.3
```

In this example, virtual router 6 is disabled. (A virtual router must be disabled before IP addresses can be added or removed from the router.) IP address 10.10.2.3 is then removed from the virtual router with the **no** form of the **vrrp address** command.

Configuring the Advertisement Interval

The advertisement interval is configurable, but all virtual routers with the same VRID must be configured with the same value. If the advertisement interval is set differently for a master router and a backup router, VRRP packets might be dropped because the newly configured interval does not match the interval indicated in the packet. The backup router then takes over and sends a gratuitous ARP, which includes the virtual router IP address and the virtual router MAC address. In addition to creating duplicate IP/MAC address messages, both routers begin forwarding packets sent to the virtual router MAC address. This results in forwarding duplicate packets.

To avoid duplicate addresses and packets, ensure to configure the same advertisement interval on both the master and the backup router.

For more information about VRRP and ARP requests, see [“ARP Requests” on page 34-8](#).

To configure the advertisement interval, use the **vrrp** command with the **interval** keyword. For example:

```
-> vrrp 6 4 disable
-> vrrp 6 4 interval 5
```

In this example, virtual router 6 is disabled. (If you are modifying an existing virtual router, the virtual router must be disabled before modification.) The **vrrp** command is then used to set the advertising interval for virtual router 6 seconds to 5 seconds.

Configuring Virtual Router Priority

VRRP functions with one master virtual router and at least one backup virtual router. A priority value determines the role each router plays. It also decides the selection of backup routers for taking over as the master router, if the master router is unavailable.

Priority values range from 1 to 254. The default priority value is 100. If a VRRP router owns the IP address of the virtual router and the IP address of the physical interface, this router functions as a virtual router master with priority value 255. The value cannot be set to 255 if the router is not the IP address owner.

If there is more than one backup router, it is necessary to configure their priorities with different values. This is done so to elect the backup router with the highest value as the master. If the priority values are the same, the backup virtual router with the highest physical interface IP address is chosen as the master.

To set the priority, use the **vrrp** command with the **priority** keyword and the desired value. For example:

```
-> vrrp 6 4 disable
-> vrrp 6 4 priority 50
```

In this example, virtual router 6 is disabled. (If you are modifying an existing virtual router, it must be disabled before it is modified.) The virtual router priority is then set to 50. Since the default priority is 100, setting the value to 50 provides the router with lower priority in the VRRP network.

Setting Preemption for Virtual Routers

When a master virtual router becomes unavailable (goes down for whatever reason), a backup router takes over. When there is more than one backup router and if their priority values are nearly equal, the skew time may not be sufficient to overcome delays caused by network traffic loads. This can cause a lower priority backup to assume control before a higher priority backup. But when the preempt mode is enabled, the higher priority backup router detects this and assumes control.

Note. In certain cases, this is not a desirable behavior, as when the original master comes back, and immediately causes all the traffic to switch back to it.

If all virtual routers have the preempt mode enabled (the default), the virtual router with the highest priority becomes the master. If the master router goes down, the highest priority backup router becomes the master. If the previous master or any other virtual router comes up with the preempt mode enabled and has a higher priority value, this router becomes the new master.

To prevent a router with a higher priority value from automatically taking control from a master router with a lower priority value, disable the preempt mode for the higher priority router. This is done by using the **no preempt** keywords with the **vrrp** command. For example:

```
-> vrrp 6 4 disable
-> vrrp 6 4 no preempt
```

Note. The virtual router that owns the IP addresses associated with the physical router always becomes the master router if it is available, regardless of the preempt mode setting and the priority values of the backup routers.

In the above example, the first command administratively disables virtual router 6. (If you are modifying an existing virtual router, it must be disabled before it is modified.). The second command disables the preempt mode for the same router. Henceforth, router 6 does not preempt another virtual router with a lower priority. For more information about priority, see [“Configuring Virtual Router Priority” on page 34-12](#).

Configuring VRRP Authentication

VRRP is designed for a range of internetworking environments that may employ different security policies. The protocol includes two authentication methods (simple clear text password and IP authentication with MD5 HMAC). *In the current release, IP authentication with MD5 HMAC is not supported.*

By default, VRRP authentication is not enabled. VRRP includes a mechanism, however, independent of whether or not authentication is configured, that denies VRRP packets from remote networks. Whenever a VRRP router receives a packet, it sets the Time To Live (TTL) to 255. This prevents the local VRRP network from accepting VRRP packets from remote networks.

When a VRRP interface receives a VRRP packet, it verifies that the TTL is 255, the VRRP version is correct, the checksum is correct, and the packet length is greater than or equal to the VRRP header. If the virtual router is configured for authentication, it will also authenticate the packet. (The authentication process is transparent to the user.)

Note. The only scenario where authentication is not recommended is an environment with minimal security risk and little chance for configuration error (such as two VRRP routers on a LAN).

Typically, simple text password authentication should be configured for VRRP. Simple text password authentication is similar to simple text authentication for the Open Shortest Path First (OSPF) routing protocol.

Simple text authentication is recommended because it protects against accidental misconfiguration of routers on a LAN and inadvertently backing up another router. If authentication is used, all virtual routers on the LAN must be configured with the same password and the password must not be the same as any significant security password.

This type of authentication is recommended when there is minimal risk of nodes on a LAN actively disrupting VRRP operation. If this type of authentication is used, the user should be aware that the clear text password is sent out frequently. It is possible for the password to be learned by a node snooping VRRP packets on the LAN; however, the simple text authentication combined with VRRP's built-in TTL check make it difficult for a VRRP packet to be sent from a remote network to disrupt VRRP operation.

To configure authentication for a virtual router, use the **authenticate** keyword and the desired password with the **vrrp** command. For example:

```
-> vrrp 6 4 disable
-> vrrp 6 4 authenticate wwwtoe
```

In this example, virtual router 6 is disabled. (If you are modifying an existing virtual router, the virtual router must be disabled before it may be modified.) The virtual router is then configured for authentication with the password **wwwtoe**. VRRP packets will be authenticated with this password.

Note. All VRRP routers on the same LAN should be configured with the same authentication setting. If authentication is enabled, all routers must have the same password.

To remove authentication from a virtual router, use the keyword with **no**. For example:

```
-> vrrp 6 4 no authenticate
```

Note that if you are modifying an existing virtual router, the virtual router must be disabled before authentication may be disabled.

Enabling/Disabling a Virtual Router

Virtual routers are disabled by default. To enable a virtual router, use the **vrrp** command with the **enable** keyword. At least one IP address must be configured for the virtual router through the **vrrp address** command. For example:

```
-> vrrp 7 3 priority 150
-> vrrp 7 3 address 10.10.2.3
-> vrrp 7 3 enable
```

In this example, a virtual router is created on VLAN 3 with a VRID of 7. An IP address is then assigned to the virtual router. The virtual router is then enabled on the switch.

To disable a virtual router, use the **disable** keyword.

```
-> vrrp 7 3 disable
```

A virtual router must be disabled before it is modified. Use the **vrrp** command to disable the virtual router first; then use the command again to modify the parameters. For example:

```
-> vrrp 7 3 disable
-> vrrp 7 3 priority 200
-> vrrp 7 3 enable
```

In this example, virtual router 7 on VLAN 3 is disabled. The virtual router is then modified to change its priority setting. (For information about configuring the priority setting, see [“Configuring Virtual Router Priority” on page 34-12.](#)) The virtual router is then re-enabled and become active on the switch.

Setting VRRP Traps

A VRRP router can generate VRRP SNMP traps for events defined in the VRRP SNMP MIB. By default traps are enabled.

In order for VRRP traps to be generated correctly, traps in general must be enabled on the switch through the SNMP CLI. See the *OmniSwitch AOS Release 6 Switch Management Guide* for more information about enabling SNMP traps globally.

To disable VRRP traps, use the **no** form of the **vrrp trap** command.

```
-> no vrrp trap
```

To re-enable traps, enter the **vrrp trap** command.

```
-> vrrp trap
```

Setting VRRP Startup Delay

After a switch reboot, the delay which is a global value, takes effect and all virtual routers remain in the **initialize** state. They remain in this state until the timer expires, at which point they negotiate to determine whether to become the master or a backup.

To set a delay to all the virtual routers from going active before their routing tables are set up, use the **vrrp delay** command. This command applies only when the switch reboots.

```
-> vrrp delay 75
```

The switch now waits 75 seconds after its reboot before it becomes available to take over as master for another router.

Note. This command applies only when the switch reboots.

Configuring Collective Management Functionality

Collective management simplifies the management and configuration tasks of either all the virtual routers on the switch or only the virtual routers in a particular virtual router group.

The following section describes the collective management functionality in detail:

Changing Default Parameter Values for all Virtual Routers

You can change the default advertising interval value of all the virtual routers on a switch using the **vrrp interval** command. For example:

```
-> vrrp interval 50
```

You can change the default priority value of all the virtual routers on a switch using the **vrrp priority** command. For example:

```
-> vrrp priority 50
```

You can change the default preempt mode of all the virtual routers on a switch using the **vrrp preempt** command. For example:

```
-> vrrp no preempt
```

These commands sets the new default values only for the virtual routers that are newly created. However, you can apply the new default value to the existing virtual routers. To apply the new default value to the existing virtual routers; first disable the virtual routers, then apply the new default value using the **vrrp set** command and enable the virtual routers again.

For example, to change the default priority value to 50 on all the existing virtual routers on a switch, enter the following:

```
-> vrrp priority 50
-> vrrp disable
-> vrrp set priority
-> vrrp enable
```

The first command configures the default priority value as 50 for all the virtual routers on the switch. The next command disables all the virtual routers on the switch. The **vrrp set** command in this sequence applies the new default priority value to the existing virtual routers. This value is applied only to those virtual routers already having the default values and not the values configured either individually or through a group. This is because the configured values take priority over the default values.

For the modified default values to effect the virtual routers which are configured with a value either individually or through a group, you can use the same command in addition with the **override** option. For example:

```
-> vrrp set priority override
```

Note. You can specify a parameter such as interval, priority, preempt or all in the **vrrp set** command to set and/or override the existing value with the new default values. By default, the option **all** is applied. The **all** option resets and/or overrides the existing advertising interval value, priority value and preempt mode with the modified default values.

The next command enables all the virtual routers on the switch except the virtual routers that are disabled individually or through a group. To enable all the virtual routers on the switch including those which are

disabled individually or through a group, you can use the same command with the **enable all** option as follows:

```
-> vrrp enable all
```

Note. This collective virtual routers management functionality does not affect the ability to change the administrative status and parameter values of an individual virtual router.

Changing Default Parameter Values for a Virtual Router Group

The virtual routers can also be grouped under a virtual router group as another way of simplifying the configuration and management tasks.

A virtual router group can be created using the **vrrp group** command as follows:

```
-> vrrp group 25
```

This command creates a virtual router group 25. Use the **no** form of the same command to delete a virtual router group. For example:

```
-> no vrrp group 25
```

Note. When a virtual router group is deleted, the virtual routers assigned to the group become unassigned. However, this does not have any impact on the virtual routers.

After creating a virtual router group, add virtual routers to the group using the **vrrp group-association** command as follows:

```
-> vrrp 10 1 group-association 25
```

The **vrrp group-association** command adds the virtual router 10 on VLAN 1 to the virtual router group 25. A virtual router need not be disabled in order to be added to a virtual router group. However, the virtual router does not adopt the default parameter values of the group until those values are applied by re-enabling the virtual router.

To remove a virtual router from a virtual router group, use the **no** form of the same command as follows:

```
-> vrrp 10 1 no group-association 25
```

A virtual router need not to be disabled to be removed from a group.

You can change the default values of the parameters like advertising interval, priority, and preempt of all the virtual routers in a virtual router group using the **vrrp group** command, as follows:

```
-> vrrp group 25 advertising interval 50 priority 50 no preempt
```

The **vrrp group** command configures the default values for advertising interval as 50 seconds, priority as 150 and preempting mode as **no preempt**. These parameters can be modified at any time. This does not affect the virtual routers in the group until you disable, then apply the group default value using the **vrrp group set** command and enable the virtual router group again.

For the modified default values to be applied to the virtual routers in a group, disable the virtual router group, then apply the group default value using the **vrrp group set** command and enable the virtual router group again. For example:

```
-> vrrp group 25 interval 50
-> vrrp group 25 disable
-> vrrp group 25 set interval
-> vrrp group 25 enable
```

The first command configures the default interval value as 50 for all the virtual routers in the virtual router group 25. The next command disables all the virtual routers in the group. **The vrrp group set** command in this sequence applies the new default interval value to all the virtual routers in the group. This value is applied only to the virtual routers in the group that already have the default value and not the values configured individually. This is because the configured values take priority over the default values.

For the modified default values to affect the virtual routers in the group, including the virtual routers that are configured with a value individually, you can use the same command in addition with the **override** option. For example:

```
-> vrrp group set interval override
```

Note. You can specify a parameter such as interval, priority, preempt or all in the **vrrp group set** command to set and/or override the existing value with the new default values. By default the option **all** is applied. The **all** option resets and/or overrides the existing advertising interval value, priority value and preempt mode with the modified default values.

The next command enables all the virtual routers in the group except the virtual routers that are disabled individually. To enable all the virtual routers in the group including the routers that are disabled individually, use the command with the **enable all** option as follows:

```
-> vrrp group 25 enable all
```

Note. Even though a virtual router can be assigned to a group, its parameter values and administrative status can still be modified individually.

Verifying the VRRP Configuration

A summary of the **show** commands used for verifying the VRRP configuration is given here:

| | |
|------------------------------------|---|
| show vrrp | Displays the virtual router configuration for all virtual routers or for a particular virtual router. |
| show vrrp statistics | Displays statistics about VRRP packets for all virtual routers configured on the switch or for a particular virtual router. |
| show vrrp track | Displays information about tracking policies on the switch. |
| show vrrp track-association | Displays the tracking policies associated with virtual routers. |
| show vrrp group | Displays the default parameter values for all the virtual router groups or for a specific virtual router group. |
| show vrrp group-association | Displays the virtual routers that are associated with a group. |

For more information about the displays that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

VRRPv3 Configuration Overview

During startup, VRRPv3 is loaded onto the switch and is enabled. Virtual routers must be configured first and enabled as described in the sections. Since VRRPv3 is implemented on multiple switches in the network, some VRRPv3 parameters must be identical across switches:

- **VRRPv3 and ACLs**

If QoS filtering rules (Access Control Lists) are configured for Layer 3 traffic on a VRRP router, all of the VRRP routers on the LAN must be configured with the same filtering rules; otherwise the security of the network is compromised. For more information about filtering, see [Chapter 40, “Configuring ACLs.”](#)

- **Conflicting VRRPv3 Parameters Across Switches**

All virtual routers with the same VRID on the LAN must be configured with the same advertisement interval and IP addresses, and authentication password. If the virtual routers are configured differently, it might result in more than one virtual router acting as the master router. This in turn would result in duplicate IP and MAC address messages as well as multiple routers forwarding duplicate packets to the virtual router MAC address. Use the [show vrrp statistics](#) command to check for conflicting parameters. For information about configuring VRRPv3 parameters, see the remaining sections of this chapter.

Basic VRRPv3 Virtual Router Configuration

At least two VRRPv3 virtual routers must be configured on the LAN—a master router and a backup router. The VRRPv3 virtual router is identified by a number called the Virtual Router ID (VRID), the VLAN on which the VRRPv3 virtual router is configured, and the IPv6 address or addresses associated with the router. Multiple VRRPv3 virtual routers can be configured on a single physical VRRP router.

Basic commands for setting up VRRPv3 virtual routers include:

```
vrrp3  
vrrp3 address
```

The next sections describe how to use these commands.

Creating/Deleting a VRRPv3 Virtual Router

To create a VRRPv3 virtual router, enter the [vrrp3](#) command with the desired VRID and the relevant VLAN ID. The VRID must be a unique number in the range from 1 to 255. The VLAN must already be created on the switch through the [vlan](#) command. For information about creating VLANs, see [Chapter 4, “Configuring VLANs.”](#) For example:

```
-> vrrp3 6 4
```

This command creates VRID 6 on VLAN 4.

When you create a new VRRPv3 virtual router, the VRID ID and a VLAN ID are *required*. Optionally, you can also specify:

- **Priority** (in the range from 1 to 255); use the **priority** keyword with the desired value. The default is 100. The IP address owner is automatically assigned a value of 255, which overrides any value already configured. See [“Configuring the VRRPv3 Virtual Router Priority” on page 34-23](#) for more information about how priority is used.

- **Preempt mode.** By default, preempt mode is enabled. Use **no preempt** to turn it off, and **preempt** to turn it back on. For more information about the preempt mode, see [“Setting Preemption for VRRPv3 Virtual Routers” on page 34-23](#).
- **Accept mode.** By default, the **accept** mode is enabled. This mode allows the master router to accept packets addressed to the IPv6 address owner as its own. Use the **no accept** mode to prevent the master router from accepting packets addressed to the IPv6 address owner.
- **Advertising interval (in centiseconds).** Use the **interval** keyword with the desired number of centiseconds for the delay in sending VRRPv3 advertisement packets. The default is 100 centiseconds. See [“Configuring the VRRPv3 Advertisement Interval” on page 34-22](#).

Note.

- The maximum number of virtual routers supported is based on the 100 centisecond interval. A smaller interval results in a relatively lesser number of virtual routers.

- The centisecond interval cannot be less than 10 centiseconds.

- **VRRP authentication.** By default, VRRP packets are not authenticated; use **authenticate** to enable simple text password authentication. A password of up to 16 characters must be entered. Use **no authenticate** to disable authentication. See [“Configuring VRRP Authentication” on page 34-13](#).

The following example creates a VRRPv3 virtual router (with VRID 7) on VLAN 2 with a priority of 75, and no preempt. VRRPv3 advertisements are sent at intervals of 200 centiseconds, and VRRP packets will be authenticated with the password **wwwtoe**:

```
-> vrrp3 7 2 priority 75 no preempt interval 200
```

Note. All VRRPv3 virtual routers with the same VRID on the same LAN must be configured with the same advertisement interval; otherwise the network can produce duplicate IPv6 or MAC address messages. These virtual routers should be configured with the same authentication setting. If authentication is enabled, all routers must have the same password.

The **vrrp3** command can also be used to specify whether the VRRPv3 virtual router is enabled or disabled (it is disabled by default). For more information about the **vrrp3** command syntax, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

To delete a VRRPv3 virtual router, use the **no** form of the **vrrp3** command with the relevant VRID and VLAN ID. For example:

```
-> no vrrp3 7 3
```

VRRPv3 virtual router 7 on VLAN 3 is deleted from the configuration. (The virtual router does not have to be disabled before you delete it.)

Specify an IPv6 Address for a VRRPv3 Virtual Router

A VRRPv3 virtual router must have a link local address. By default, the virtual router link local address is created based on the virtual router MAC address and need not be configured. Additional IPv6 addresses can be configured for a virtual router and these addresses must be within the subnet of an address configured on the interface. To specify an IPv6 address for a VRRPv3 virtual router, use the **vrrp3 address** command and the relevant IPv6 address. For example:

```
-> vrrp3 6 4 address fe80::200:5eff:fe00:20a
-> vrrp3 6 4 enable
```

In the above example, the **vrrp3 address** command specifies that VRRPv3 virtual router 6 on VLAN 4 is used to back up IPv6 address `fe80::200:5eff:fe00:20a`. The virtual router is then enabled with the **vrrp3** command.

If a virtual router is to be the IP address owner, then all addresses on the virtual router must match an address on the switch interface. This includes the virtual router's link local address. In other words, a virtual router cannot be the IP address owner if its link local address does not match the interface link local address.

To remove an IPv6 address from a virtual router, use the **no** form of the **vrrp3 address** command. For example:

```
-> vrrp3 6 4 disable
-> vrrp3 6 4 no address fe80::200:5eff:fe00:20a
```

In this example, VRRPv3 virtual router 6 is disabled. (A VRRPv3 virtual router must be disabled before IPv6 addresses can be added or removed from the router.) IP address `fe80::200:5eff:fe00:20a` is then removed from the virtual router with the **no** form of the **vrrp3 address** command.

Configuring the VRRPv3 Advertisement Interval

The advertisement interval is configurable, but all virtual routers with the same VRID must be configured with the same value. If the advertisement interval is set differently for a master router and a backup router, VRRPv3 packets might be dropped because the newly configured interval does not match the interval indicated in the packet. The backup router then takes over and sends a neighbor advertisement, which includes the virtual router IP address and the virtual router MAC address. In addition to creating duplicate IP/MAC address messages, both routers begin forwarding packets sent to the virtual router MAC address. This results in forwarding duplicate packets.

To avoid duplicate addresses and packets, ensure to configure the same advertisement interval on both the master and the backup router.

To configure the advertisement interval, use the **vrrp3** command with the **interval** keyword. For example:

```
-> vrrp3 6 4 disable
-> vrrp3 6 4 interval 500
```

In this example, VRRPv3 virtual router 6 is disabled. (If you are modifying an existing virtual router, the virtual router must be disabled before it is modified.) The **vrrp3** command is then used to set the advertising interval for virtual router 6 to 500 centiseconds.

Configuring the VRRPv3 Virtual Router Priority

VRRPv3 functions with one master virtual router and at least one backup virtual router. A priority value determines the role each router plays. If the master router is unavailable, the priority value decides the selection of backup routers for taking over as the master router.

Priority values range from 1 to 254. A value of 255 indicates that the virtual router owns the IPv6 address; that is, the router contains the real physical interface to which the IPv6 address is assigned. The default priority value is 100; however, if the router is the IPv6 address owner, the switch sets the priority value to 255. If the router is not the IPv6 address owner, the priority value cannot be set to 255.

The IPv6 address owner will always be the master router if available. If more than one backup router is configured, their priority values must be configured with different values, so that the backup with the higher value takes over for the master. The priority parameter can be used to control the order in which backup routers takes over for the master. If priority values are the same, any backup takes over for master.

The switch sets the priority value to zero in the last VRRPv3 advertisement packet before a master router is disabled (see [“Enabling/Disabling a VRRPv3 Virtual Router” on page 34-25](#)).

Also, if a router is the IPv6 address owner and the priority value is not set to 255, the switch sets its priority to 255 when the router is enabled.

To set the priority, use the **vrrp3** command with the **priority** keyword and the desired value. For example:

```
-> vrrp3 6 4 disable
-> vrrp3 6 4 priority 50
```

In this example, VRRPv3 virtual router 6 is disabled. (If you are modifying an existing virtual router, the virtual router must be disabled before it is modified.) The virtual router priority is then set to 50. The priority value is relative to the priority value configured for other virtual routers backing up the same IPv6 address. Since the default priority is 100, setting the value to 50 would typically provide a router with lower priority in the VRRPv3 network.

Setting Preemption for VRRPv3 Virtual Routers

When a VRRPv3 master virtual router becomes unavailable (goes down for whatever reason), a backup router takes over. When there is more than one backup router and if the backup routers have priority values that are nearly equal, the skew time might not be sufficient to overcome delays caused by network traffic loads and a lower priority backup may assume control before a higher priority backup. But when the preempt mode is enabled the higher priority backup router detects this and assumes control.

By default, VRRPv3 virtual routers are allowed to preempt each other. If the master router becomes unavailable, the virtual router with the highest priority takes over. The preempt mode can be disabled so that any backup router that takes over when the master is unavailable is not preempted by a backup with a higher priority.

Note. The VRRPv3 virtual router that owns the IPv6 addresses associated with the physical router always becomes the master router if it is available regardless of the preempt mode setting and the priority values of the backup routers.

To disable preemption for a VRRPv3 virtual router, use the **vrrp3** command with the **no preempt** keywords. For example:

```
-> vrrp3 6 4 disable
-> vrrp3 6 4 no preempt
```

In this example, virtual router 6 is disabled. (If you are modifying an existing virtual router, the virtual router must be disabled before it is modified.) The virtual router is then configured to disable preemption. If this virtual router takes over for an unavailable router, a router with a higher priority will not be able to preempt it. For more information about priority, see [“Configuring the VRRPv3 Virtual Router Priority” on page 34-23](#).

Configuring VRRP Authentication

VRRP is designed for a range of internetworking environments that may employ different security policies. The protocol includes two authentication methods (simple clear text password and IP authentication with MD5 HMAC). *In the current release, IP authentication with MD5 HMAC is not supported.*

By default, VRRP authentication is not enabled. VRRP includes a mechanism, however, independent of whether or not authentication is configured, that denies VRRP packets from remote networks. Whenever a VRRP router receives a packet, it sets the Time To Live (TTL) to 255. This prevents the local VRRP network from accepting VRRP packets from remote networks.

When a VRRP interface receives a VRRP packet, it verifies that the TTL is 255, the VRRP version is correct, the checksum is correct, and the packet length is greater than or equal to the VRRP header. If the virtual router is configured for authentication, it will also authenticate the packet. (The authentication process is transparent to the user.)

Note. The only scenario where authentication is not recommended is an environment with minimal security risk and little chance for configuration error (such as two VRRP routers on a LAN).

Typically, simple text password authentication should be configured for VRRP. Simple text password authentication is similar to simple text authentication for the Open Shortest Path First (OSPF) routing protocol.

Simple text authentication is recommended because it protects against accidental misconfiguration of routers on a LAN and inadvertently backing up another router. If authentication is used, all virtual routers on the LAN must be configured with the same password and the password must not be the same as any significant security password.

This type of authentication is recommended when there is minimal risk of nodes on a LAN actively disrupting VRRP operation. If this type of authentication is used, the user should be aware that the clear text password is sent out frequently. It is possible for the password to be learned by a node snooping VRRP packets on the LAN; however, the simple text authentication combined with VRRP's built-in TTL check make it difficult for a VRRP packet to be sent from a remote network to disrupt VRRP operation.

To configure authentication for a virtual router, use the **authenticate** keyword and the desired password with the **vrrp** command. For example:

```
-> vrrp 6 4 disable
-> vrrp 6 4 authenticate wwwtoe
```

In this example, virtual router 6 is disabled. (If you are modifying an existing virtual router, the virtual router must be disabled before it may be modified.) The virtual router is then configured for authentication with the password **wwwtoe**. VRRP packets will be authenticated with this password.

Note. All VRRP routers on the same LAN should be configured with the same authentication setting. If authentication is enabled, all routers must have the same password.

To remove authentication from a virtual router, use the keyword with **no**. For example:

```
-> vrrp 6 4 no authenticate
```

Note that if you are modifying an existing virtual router, the virtual router must be disabled before authentication may be disabled.

Enabling/Disabling a VRRPv3 Virtual Router

VRRPv3 virtual routers are disabled by default. To enable a virtual router, use the **vrrp3** command with the **enable** keyword. For example:

```
-> vrrp3 7 3
-> vrrp3 7 3 enable
```

In this example, a VRRPv3 virtual router is created on VLAN 3 with a VRID of 7. An IPv6 address is then assigned to the virtual router. The virtual router is then enabled on the switch.

To disable a VRRPv3 virtual router, use the **disable** keyword.

```
-> vrrp 7 3 disable
```

A VRRPv3 virtual router must be disabled before it is modified. Use the **vrrp3** command to disable the virtual router first; then use the command again to modify the parameters. For example:

```
-> vrrp3 7 3 disable
-> vrrp3 7 3 priority 200
-> vrrp3 7 3 enable
```

In this example, VRRPv3 virtual router 7 on VLAN 3 is disabled. The VRRPv3 virtual router is then modified to change its priority setting. (For information about configuring the priority setting, see [“Configuring the VRRPv3 Virtual Router Priority” on page 34-23](#).) The virtual router is then re-enabled and becomes active on the switch.

Setting VRRPv3 Traps

A VRRPv3 router can generate VRRPv3 SNMP traps for events defined in the VRRPv3 SNMP MIB. By default traps are enabled.

In order for VRRPv3 traps to be generated correctly, traps in general must be enabled on the switch through the SNMP CLI. See the *OmniSwitch AOS Release 6 Switch Management Guide* for more information about enabling SNMP traps globally.

To disable VRRPv3 traps, use the **no** form of the **vrrp3 trap** command.

```
-> no vrrp3 trap
```

To re-enable traps, enter the **vrrp3 trap** command:

```
-> vrrp3 trap
```

Verifying the VRRPv3 Configuration

A summary of the **show** commands used for verifying the VRRPv3 configuration is given here:

- | | |
|-------------------------------------|--|
| show vrrp3 | Displays the VRRPv3 virtual router configuration for all virtual routers or for a particular virtual router. |
| show vrrp3 statistics | Displays statistics about VRRPv3 packets for all VRRPv3 virtual routers configured on the switch or for a particular virtual router. |
| show vrrp3 track-association | Displays the tracking policies associated with VRRPv3 virtual routers. |

For more information about the displays that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Creating Tracking Policies

To create a tracking policy, use the **vrrp track** command and specify the amount to decrease the priority of the virtual router and the slot/port, IP address, or IP interface name to be tracked. For example:

```
-> vrrp track 3 enable priority 50 address 20.1.1.3
```

In this example, a tracking policy ID (3) is created and enabled for IP address 20.1.1.3. If this address becomes unreachable, a virtual router associated with this track ID has its priority decremented by 50. The **enable** keyword administratively activates the tracking policy, but the policy does not take effect until it is associated with one or more virtual routers (see the next section).

Similarly, to create a tracking policy ID (3) for IPv6 address 213:100:1::56, use the following command:

```
-> vrrp track 3 enable priority 50 address 213:100:1::56
```

If this address becomes unreachable, a virtual router associated with this track ID has its priority decremented by 50.

Note the following:

- A virtual router must be administratively disabled before a tracking policy for the virtual router can be added.
- VRRP tracking does not override IP address ownership (the IP address owner will always have priority to become master, if it is available).

Associating a Tracking Policy with a VRRPv2/VRRPv3 Virtual Router

To associate a tracking policy with a virtual router, use the **vrrp track-association** command with the tracking policy ID number. In this example, virtual router 6 on VLAN 4 is disabled first so that tracking policy 3 can be associated with it:

```
-> vrrp 6 4 disable  
-> vrrp 6 4 track-association 3
```

When the virtual router is re-enabled, tracking policy 3 is used for that virtual router.

A tracking policy must not be associated with a virtual router on the same port or interface. For example:

```
-> ip interface vlan-4 address 10.1.1.1 vlan 4  
-> vrrp track 2 ipv4-interface vlan-4  
-> vrrp 5 4 track-association 2
```

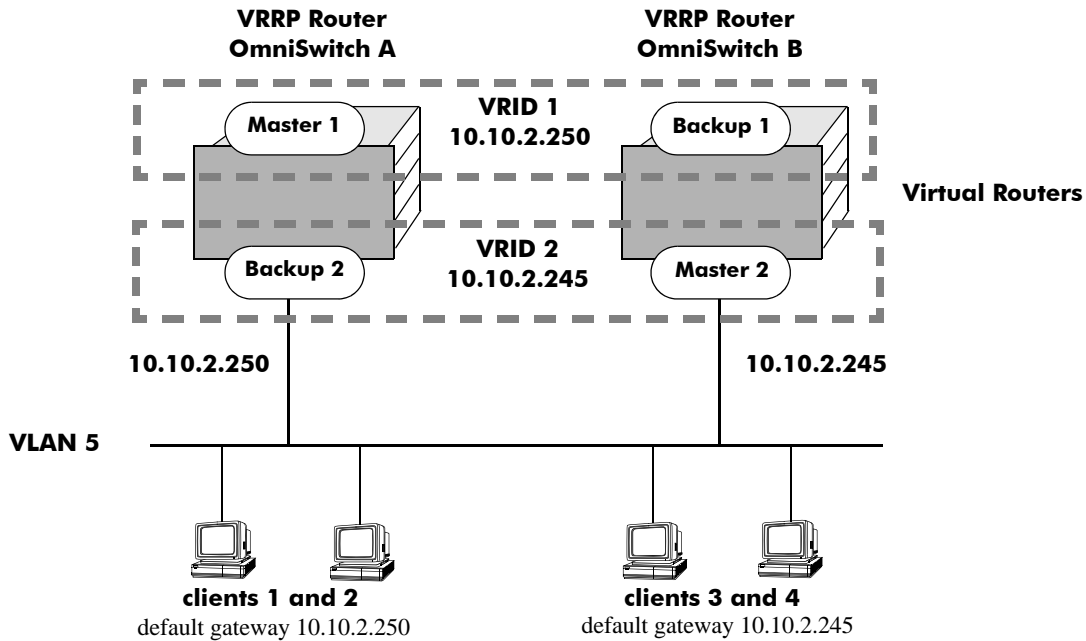
This configuration is allowed but does not affect. If the associated interface goes down, the virtual router goes down as well and the tracking policy is not applied.

Note. A master and a backup virtual router must not track the same IP address. Else, when the IP address becomes unreachable, the priority of both virtual routers is decremented, and the backup can temporarily take over if the master discovers that the IP address is unreachable before the backup.

It is recommended not to configure the same IP address tracking policies on physical VRRP routers that back up each other; otherwise, the priority is decremented for both master and backup when the entity being tracked goes down.

VRRP Application Example

In addition to providing redundancy, VRRP can assist in load balancing outgoing traffic. The following figure shows two virtual routers with their hosts splitting traffic between them. Half of the hosts are configured with a default route to IP address (10.10.2.250) of the virtual router 1, and the other half are configured with a default route to IP address (10.10.2.245) of the virtual router 2.



VRRP Redundancy and Load Balancing

The CLI commands used to configure this setup are as follows:

1 First, create two virtual routers for VLAN 5. (VLAN 5 must already be created and available on the switch.)

```
-> vrrp 1 5
-> vrrp 2 5
```

2 Configure the IP addresses for each virtual router.

```
-> vrrp 1 5 ip 10.10.2.250
-> vrrp 2 5 ip 10.10.2.245
```

3 Enable the virtual routers.

```
-> vrrp 1 5 enable
-> vrrp 2 5 enable
```

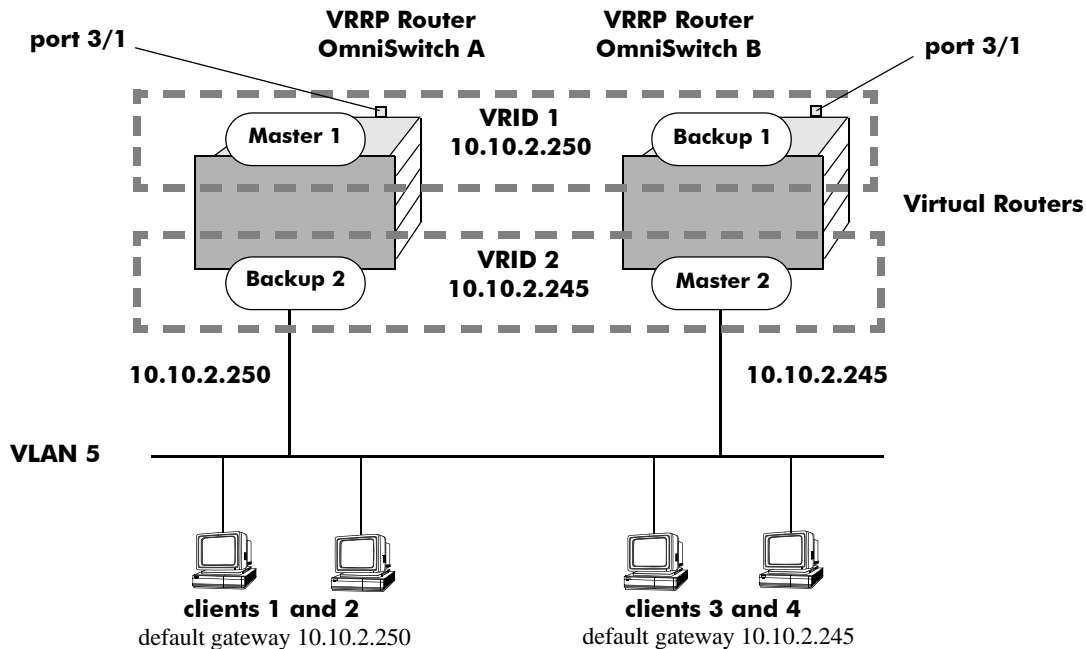
Note. The same VRRP configuration must be set up on each switch. The VRRP router that contains, or owns, the IP address automatically becomes the master for that virtual router. If the IP address is a virtual address, the virtual router with the highest priority becomes the master router.

In this scenario, the master of VRID 1 responds to ARP requests for IP address A using the virtual router MAC address for VRID 1 (00:00:5E:00:01:01). OmniSwitch 1 is the master for VRID 1 since it contains the physical interface to which 10.10.2.250 is assigned. If OmniSwitch A must become unavailable, OmniSwitch B becomes the master for VRID 1.

In the same way, the master of VRID 2 responds to ARP requests for IP address B using the virtual router MAC address for VRID 2 (00:00:5E:00:01:02). OmniSwitch B is the master for VRID 2 since it contains the physical interface to which 10.10.2.245 is assigned. If OmniSwitch B must become unavailable, OmniSwitch A becomes the master for 10.10.2.245. This configuration provides uninterrupted service for the end hosts.

VRRP Tracking Example

The following figure shows two VRRP routers with two virtual routers backing up one IP address on each VRRP router respectively. Virtual router 1 serves as the default gateway on OmniSwitch A for clients 1 and 2 through IP address 10.10.2.250. Virtual router 2 serves as default gateway on OmniSwitch B for clients 3 and 4 through IP address 10.10.2.245. For example, if the port that provides access to the Internet on OmniSwitch A fails, virtual router 1 continues to be the default router for clients 1 and 2, but clients 1 and 2 will not be able to access the Internet.



VRRP Tracking Example

In this example, the master for virtual router 1 has a priority of 100 and the backup for virtual router 1 has a priority of 75. The virtual router configuration for VRID 1 and 2 on VRRP router A is as follows:

```
-> vrrp 1 5 priority 100 preempt
-> vrrp 2 5 priority 75
```

The virtual router configuration for VRID 1 and 2 on VRRP router B is as follows:

```
-> vrrp 1 5 priority 75
-> vrrp 2 5 priority 100 preempt
```

To ensure workstation clients 1 and 2 have connectivity to the internet, configure a tracking policy on VRRP router A to monitor port 3/1 and associate the policy with VRID 1.

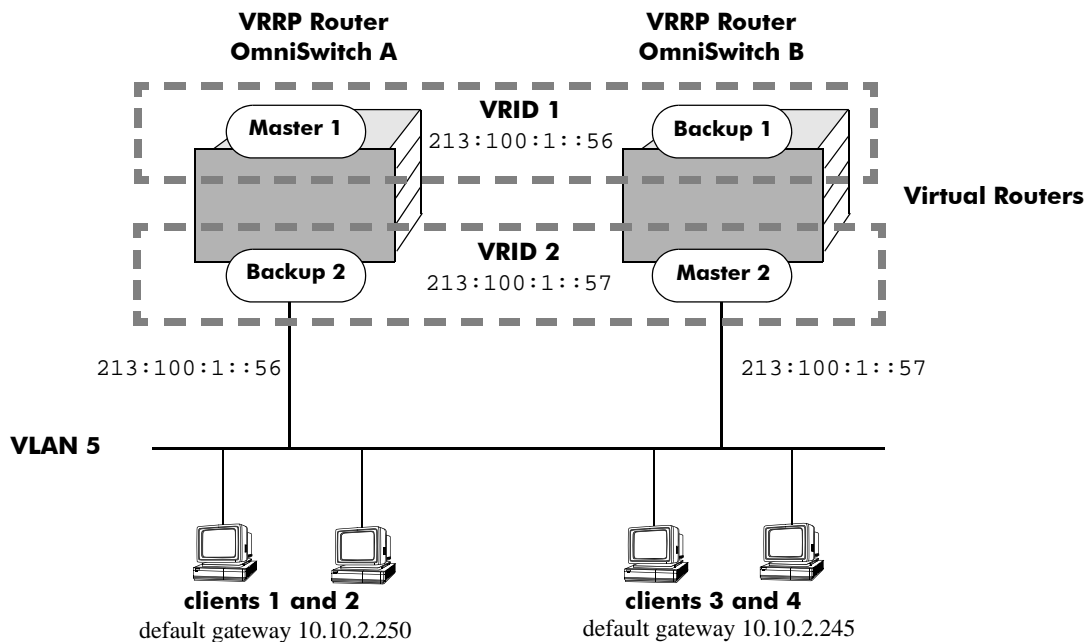
```
-> vrrp track 1 enable priority 50 port 3/1
-> vrrp 1 5 track-association 1
```

If port 3/1 on VRRP router A goes down, the master for virtual router A continue to function but workstation clients 1 and 2 will not be able access the Internet. With this tracking policy enabled, the priority of the master router 1 is temporarily decremented to 50, allowing backup router 1 to take over and provide connectivity for the workstations. When port 3/1 on VRRP router A comes backup, master 1 takes over again.

Note. Preempt must be set on switch A virtual router 1, and switch B virtual router 2 for the correct master to assume control once their respective ports 3/1 return to viability. In our example, once port 3/1 on switch A is functioning again, switch A must reestablish itself as the master. See [“Setting Preemption for Virtual Routers”](#) on page 34-13 for more information about enabling preemption.

VRRPv3 Application Example

In addition to providing redundancy, VRRPv3 can assist in load balancing outgoing traffic. The following figure shows two virtual routers with their hosts splitting traffic between them. Half of the hosts are configured with a default route to IPv6 address (213:100:1::56) of the virtual router 1. The other half are configured with a default route to IPv6 address (213:100:1::57) of the virtual router 2.



VRRPv3 Redundancy and Load Balancing

The CLI commands used to configure this setup are as follows:

- 1 First, create two VRRPv3 virtual routers for VLAN 5. (VLAN 5 must already be created and available on the switch.)

```
-> vrrp3 1 5
-> vrrp3 2 5
```

- 2 Configure the IPv6 addresses for each VRRPv3 virtual router.

```
-> vrrp3 1 5 address 213:100:1::56
-> vrrp3 2 5 address 213:100:1::57
```

- 3 Enable the VRRPv3 virtual routers.

```
-> vrrp3 1 5 enable
-> vrrp3 2 5 enable
```

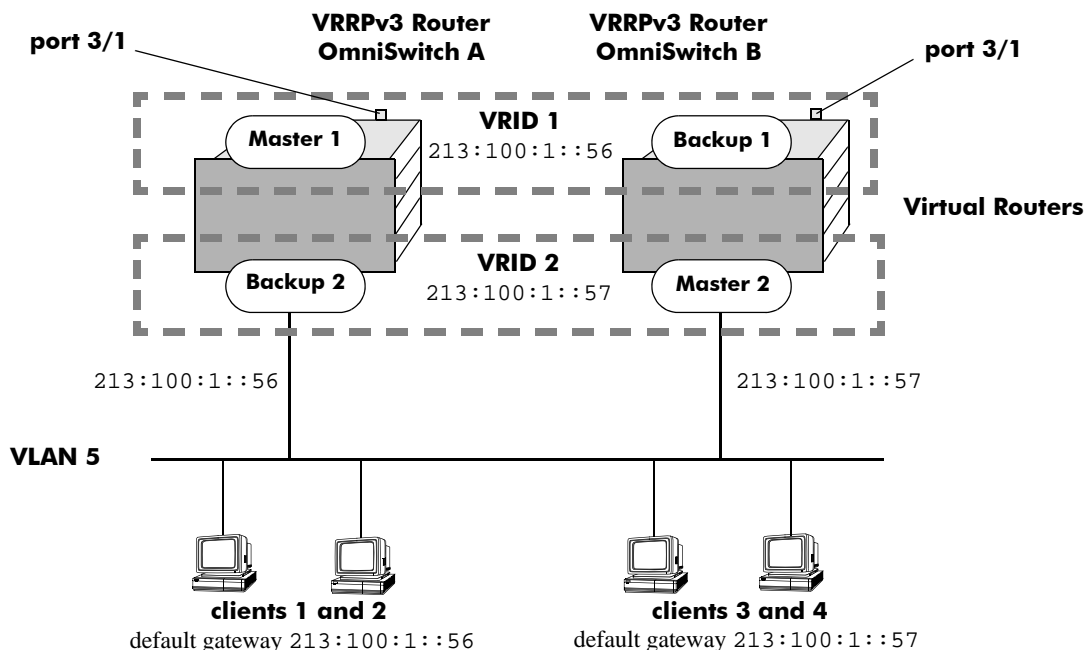

Note. The same VRRPv3 configuration must be set up on each switch. The VRRPv3 router that contains, or owns, the IPv6 address automatically becomes the master for that virtual router. If the IPv6 address is a virtual address, the virtual router with the highest priority becomes the master router.

In this scenario, the master of VRID 1 responds to neighbor solicitation with a neighbor advertisement for IPv6 address A using the virtual router MAC address for VRID 1 (00:00:5E:00:02:01). OmniSwitch 1 is the master for VRID 1 since it contains the physical interface to which 213:100:1::56s assigned. If OmniSwitch A must become unavailable, OmniSwitch B becomes master for VRID 1.

In the same way, the master of VRID 2 responds to neighbor solicitation for IPv6 address B using the virtual router MAC address for VRID 2 (00:00:5E:00:02:02). OmniSwitch B is the master for VRID 2 since it contains the physical interface to which 213:100:1::57 is assigned. If OmniSwitch B must become unavailable, OmniSwitch A becomes the master for 213:100:1::57. This configuration provides uninterrupted service for the end hosts.

VRRPv3 Tracking Example

The figure below shows two VRRPv3 routers with two virtual routers backing up one IPv6 address on each VRRPv3 router respectively. Virtual router 1 serves as the default gateway on OmniSwitch A for clients 1 and 2 through IPv6 address 213:100:1::56. For example, if the port that provides access to the Internet on OmniSwitch A fails, virtual router 1 continues to be the default router for clients 1 and 2, but clients 1 and 2 will not be able to access the Internet.



VRRPv3 Tracking Example

In this example, the master for virtual router 1 has a priority of 100 and the backup for virtual router 1 has a priority of 75. The virtual router configuration for VRID 1 and 2 on VRRPv3 router A is as follows:

```
-> vrrp3 1 5 priority 100 preempt
-> vrrp3 2 5 priority 75
```

The virtual router configuration for VRID 1 and 2 on VRRPv3 router B is as follows:

```
-> vrrp3 1 5 priority 75
-> vrrp3 2 5 priority 100 preempt
```

To ensure workstation clients 1 and 2 have connectivity to the internet, configure a tracking policy on VRRPv3 router A to monitor port 3/1 and associate the policy with VRID 1.

```
-> vrrp3 track 1 enable priority 50 port 3/1
-> vrrp3 1 5 track-association 1
```

If port 3/1 on VRR3 router A goes down, the master for virtual router A is still functioning, but workstation clients 1 and 2 will not be able to get to the Internet. With this tracking policy enabled, however, master router 1's priority will be temporarily decremented to 50, allowing backup router 1 to take over and provide connectivity for those workstations. When port 3/1 on VRRPv3 router A comes backup, master 1 takes over again.

Note. Preempt must be set on switch A virtual router 1, and switch B virtual router 2 for the correct master to assume control once their respective ports 3/1 return to viability. In our example, once port 3/1 on switch A is functioning again, switch A must reestablish itself as the master. See [“Setting Preemption for Virtual Routers” on page 34-13](#) for more information about enabling preemption.

35 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).
- **Host Integrity Check (HIC)**—An integrated solution for device integrity verification. This solution consists of the HIC server, a permanent or web-based download-able agent to verify host compliance, and User Network Profiles (UNP). HIC is triggered when a UNP is applied to a device and HIC is enabled for the UNP.

Note. For an enhanced solution using the ClearPass server and posture checking please refer to the BYOD section.

- **User Network Profiles (UNP)**—One of the configurable options of a device classification policy is to classify a device with a UNP. When the policy applies the UNP to one or more devices, the UNP determines the VLAN assignment for the device, whether or not HIC is required for the device, and if any QoS access control list (ACL) policies are applied to the device.
- **Virtual Network Profile (VNP)** - Also referred to as the **Universal Network Profile (UNP)**, it provides a method for dynamically assigning network devices to VLAN domains. A profile consists of configurable attributes that help to define a group of users or devices that have similar requirements for access to network resources. A device sending traffic that matches such attributes is then assigned to a VLAN associated with the UNP. The UNP may also specify a QoS policy list that is subsequently applied to device traffic associated with the UNP VLAN. For more information on UNP commands, see *OmniSwitch AOS Release 6 CLI Reference Guide* and for UNP configuration, see chapter “Configuring Universal Network Profiles”.

- **Bring Your Own Device (BYOD) - OmniSwitch / UPAM or ClearPass Integration:** Guest users and user devices information can be allowed to access specific network resources. BYOD support provides restricted access to the network so that the end user device can be validated, user roles identified, compliance checked, and have the correct access policies applied. The OmniSwitch leverages the Access Guardian features along with the OmniVista Unified Policy Access Manager (UPAM) or the ClearPass Policy Manager to provide the overall BYOD solution. See the “[Bring Your Own Device \(BYOD\) Overview](#)” on page 35-66 for information about the Access Guardian UPAM or ClearPass solution. For additional information refer to the following:
 - OmniAccess WLAN documentation.
 - OmniVista Unified Policy Access Manager documentation for in-depth OmniSwitch and server configuration requirements.
 - ClearPass Policy Manager documentation for in-depth server configuration and licensing requirements.
 - ClearPass and OmniSwitch Configuration Video.

Note. Find the ClearPass and OmniSwitch Configuration Videos on YouTube at the following location:

<https://www.youtube.com/watch?v=PyueDr-GAFM&list=PLrzAZN530GJ8kfUJCNSjIhJW6cAV5AACb>

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 AOS Release 6 CLI Reference Guide*.

The following information and procedures are included in this chapter:

- “[Quick Steps for Configuring Access Guardian](#)” on page 35-7
- “[Access Guardian Overview](#)” on page 35-14.
- “[Interaction With Other Features](#)” on page 35-25
- “[Setting Up Port-Based Network Access Control](#)” on page 35-26
- “[Configuring Access Guardian Policies](#)” on page 35-31
- “[Configuring 802.1x Authentication Bypass](#)” on page 35-40
- “[Configuring Captive Portal Authentication](#)” on page 35-42
- “[Configuring Host Integrity Check](#)” on page 35-49
- “[Configuring User Network Profiles](#)” on page 35-51
- “[OmniAccess Stellar AP Integration](#)” on page 35-55
- “[Verifying Access Guardian Users](#)” on page 35-62
- “[Verifying the Access Guardian Configuration](#)” on page 35-65
- “[Bring Your Own Device \(BYOD\) Overview](#)” on page 35-66

- [“Zero Configuration Networking \(mDNS and SSDP\)” on page 35-77](#)

For more information about configuring 802.1X on switch ports, see [Chapter 37, “Configuring 802.1X.”](#)

Access Guardian Specifications

| | |
|--|--|
| RFCs Supported | <p>RFC 2284–PPP Extensible Authentication Protocol (EAP)</p> <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 2869–RADIUS Extensions</p> <p>RFC 3576--Change of Authorization-Request (COA) and Disconnect request (DM) for BYOD. RFC support is limited to ClearPass solution</p> |
| IEEE Standards Supported | <p>IEEE 802.1X-2001–Standard for Port-based Network Access Control</p> <p>802.1X RADIUS Usage Guidelines</p> |
| Platforms Supported | OmniSwitch 6350, 6450 |
| Number of Host Integrity Check servers per switch | 1 |
| Number of servers allowed in the Host Integrity Check exception list | 4 |
| Maximum number of hosts processed through Host Integrity Check | 256 |
| Number of QoS policy lists per User Network Profile | 1 |
| Maximum number of servers that can be specified for authentication and accounting (802.1x and MAC) | 5 (Including the backup servers) |
| Average number of users allowed to login to Captive portal Web pages at a time. | 20 |
| BYOD Solution Server | ClearPass Policy Manager (CPPM) |
| mDNS GRE Tunnel Supported Protocol | IPv4 |

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 |
| Bypass 802.1x authentication for supplicants; perform MAC authentication first. | 802.1x supplicant bypass | disable |
| Allow supplicant authentication of MAC authenticated clients depending on MAC authentication outcome and 'allow-eap' configuration. | 802.1x non-supplicant allow-eap | none (Only MAC authentication is performed; classification with non-supplicant policies) |
| 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.) |

| Description | Keyword | Default |
|---|--|---|
| 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 |
| Transparent forwarding of 802.1x frames through switch | 802.1x pass-through | disable |
| Transparent forwarding of Captive Portal data through bridge switch | captive portal pass-through | disable |
| Bypass 802.1x authentication for supplicants; perform MAC authentication first. | 802.1x supplicant bypass | disable |

| Description | Keyword | Default |
|---|--|---|
| Allow supplicant authentication of MAC authenticated clients depending on MAC authentication outcome and 'allow-eap' configuration. | 802.1x non-supplicant allow-eap | none (Only MAC authentication is performed; classification with non-supplicant policies) |
| 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 authenticates and classifies 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 authenticates and classifies 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 associate a UNP with maximum ingress and egress bandwidth along with maximum default depth, use the **aaa user-network-profile** command with **maximum-ingress-bandwidth**, **maximum-egress-bandwidth**, and **maximum-default-depth** parameters.

```
-> aaa user-network-profile name "profile1" vlan 50 maximum-ingress-bandwidth 1024 maximum-egress-bandwidth 256 maximum-default-depth 128
```

- 4 To configure an Access Guardian Captive Portal policy that classifies web-based clients, use the **802.1x captive-portal policy authentication** command.

Note. 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
```

- 5 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
```

- 6 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
```

- 7 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
```

- 8 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-suppliant** command:

```

-> show 802.1x non-suppliant
Slot  MAC                Authentication  Classification Vlan
Port  Address              Status         Policy          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 AOS Release 6 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
```

- 2 To enable the Host Integrity Check option for a UNP, use the **aaa user-network-profile** command with the **hic enable** parameter.

```
-> aaa user-network-profile name guest_user vlan 500 hic enable
```

- 3 To associate a UNP with maximum ingress and egress bandwidth along with maximum default depth, use the **aaa user-network-profile** command with **maximum-ingress-bandwidth**, **maximum-egress-bandwidth**, and **maximum-default-depth** parameters.

```
-> aaa user-network-profile name "profile1" vlan 50 maximum-ingress-bandwidth
1024 maximum-egress-bandwidth 256 maximum-default-depth 128
```

- 4 To assign a list of QoS policies to a UNP, use the **aaa user-network-profile** command with the **policy-list-name** parameter. Note that the policy list specified must already exist in the switch configuration.

```
-> aaa user-network-profile name guest_user vlan 500 policy-list name temp_rules
```

- 5 To configure an Access Guardian device classification policy to apply a user profile, use the **802.1x supplicant policy authentication**, **802.1x non-suppliant policy authentication**, **802.1x captive-portal policy authentication**, or **802.1x non-suppliant policy** command with the **user-network-profile** parameter. For example:

```
-> 802.1x 1/10 supplicant policy authentication user-network-profile guest_user
```

Note. Verify the UNP configuration using the `show aaa user-network-profile` command:

```
-> show aaa user-network-profile
Role Name          Vlan  HIC  Policy List Name
-----+-----+-----+-----
                guest-user  500 Yes  temp_rules
                accounting  20  No   acct_rules
```

To verify the UNP configuration for a device classification policy, use 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:

-> show 802.1x device classification policies
Device classification policies on 802.1x port 1/10
Supplicant:
  authentication:
    pass: UNP guest-user, block
    fail: block
  Non-Supplicant:
    block (default)
  Captive Portal:
    authentication:
      pass: default-vlan (default)
      fail: block (default)
```

See the *OmniSwitch AOS Release 6 CLI Reference Guide* for information about the fields in this display.

See “[Configuring User Network Profiles](#)” on page 35-51 for more information about configuring profiles.

Quick Steps for Configuring User Network Profile Mobile Rules

The Group Mobility device classification policy uses VLAN mobile rules and User Network Profile (UNP) mobile rules to determine the VLAN assignment for host devices. The following quick steps provide a brief tutorial for configuring UNP mobile rules:

- 1 To configure a MAC address UNP mobile rule, use the `aaa classification-rule mac-address` command.

```
-> aaa classification-rule mac-address 00:00:2a:33:44:01 user-network-profile
name accounting
```

- 2** To configure a UNP mobile rule for a range of MAC addresses, use the **aaa classification-rule mac-address-range** command.

```
-> aaa classification-rule mac-address-range 00:00:2a:33:44:01 00:00:2a:33:44:10
user-network-profile name accounting
```

- 3** To configure an IP address UNP mobile rule, use the **aaa classification-rule ip-address** command.

```
-> aaa classification-rule ip-address 198.4.21.1 255.255.0.0 user-network-
profile name marketing
```

- 4** To configure an Access Guardian Group Mobility device classification policy to authenticate and classify devices using UNP mobile rules, use the **802.1x supplicant policy authentication**, **802.1x non-supplicant policy authentication**, **802.1x captive-portal policy authentication**, or **802.1x non-supplicant policy** command with the **group-mobility** parameter. For example:

```
-> 802.1x 6/1 supplicant policy authentication pass group-mobility default-vlan
fail captive-portal
```

Note. If the default VLAN for port is same as User Network Profile (UNP) VLAN, then the UNP QoS Policy List is not applied. Default VLAN of the port must be different from that of the UNP VLAN.

Verify the UNP mobile rule configuration using the **show aaa classification-rule** command:

```
-> show aaa classification-rule mac-rule
MAC Address          User Network Profile Name
-----+-----
00:1a:a0:b1:fa:e5    guest_user
00:b0:d0:2a:0e:2e    acct_user
00:b0:d0:2a:11:60    engr_user

-> show aaa classification-rule mac-range-rule
Low MAC Address      High MAC Address  User Network Profile Name
-----+-----+-----
00:1a:a0:b1:fa:10    00:1a:0a:b1:fa:20  guest_user
00:b0:d0:2a:0e:2e    00:b0:d0:2a:0e:3a  acct_user
00:b0:d0:2a:11:60    00:b0:d0:2a:11:70  engr_user

-> show aaa classification-rule ip-net-rule
IP Addr             IP Mask           User Network Profile Name
-----+-----+-----
198.4.21.1          255.255.0.0      guest_user
10.1.1.1             255.0.0.0         acct_user
20.2.2.1             255.0.0.0         engr_user
```

See the *OmniSwitch AOS Release 6 CLI Reference Guide* for information about the fields in this display.

See “Configuring User Network Profile Mobile Rules” on page 35-54 for more information.

Quick Steps for Configuring Host Integrity Check

The Host Integrity Check (HIC) feature is a configurable option for Access Guardian User Network Profiles (UNP). However, other configuration tasks are required to make the HIC process available through the switch. The following quick steps provide a brief tutorial for configuring HIC (InfoExpress CyberGatekeeper) server information and the global HIC status and parameter values for the switch:

- 1 Configure the name, IP address, and shared secret of the InfoExpress CyberGatekeeper server using the `aaa hic server-name` command. This step is required before HIC can be enabled for the switch.

```
-> aaa hic server-name hic_srv1 ip-address 2.2.2.1 key wwwtoe role primary
```

- 2 Enable the HIC feature for the switch using the `aaa hic` command.

```
-> aaa hic enable
```

- 3 Enable the HIC option for the UNP using the `aaa user-network-profile` command.

```
-> aaa user-network-profile name guest_user vlan 500 hic enable
```

- 4 *Optional.* Configure a server name and IP address entry for the HIC exception list using the `aaa classification-rule mac-address` command.

```
-> aaa hic allowed-name rem_srv1 ip-address 10.1.1.1
```

- 5 *Optional.* Configure the URL for the web-agent download server using the `aaa hic web-agent-url` command.

```
-> aaa hic web-agent-url http://10.10.10.10:2146
```

- 6 *Optional.* Configure the proxy port number for the host device using the `aaa hic custom-proxy-port` command.

```
-> aaa hic custom-proxy-port 8878
```

Note. Verify the HIC configuration for the switch using the `show aaa classification-rule` command:

```
-> show aaa hic
HIC Global Status: Enabled
HIC Allowed 1:      rem-srv1
HIC Web Agent URL: http://100.100.100.100:8080/CGAgentLauncher.htm
HIC Proxy Port:    8383
HIC Reconnect-timer: 16
HIC Server-fail-mode: Passthrough
```

To verify the HIC InfoSys CyberGatekeeper server information configured for the switch, use the `show aaa classification-rule` command:

```
-> show aaa hic server
Server
Name                IP Address      UDP Port      Server Role      Server Connection      Server Status
-----+-----+-----+-----+-----+-----+-----
      hic-srv1         10.2.2.2        11707        Primary          Active              Down
      hic              2.2.2.1         11707        Backup           Inactive            Down
```

To display the HIC status for host devices, use the `show aaa hic host` command:

```
-> show aaa hic host
  HIC Host MAC          Status
-----+-----
00:1a:a0:b1:fa:e5      Successful
00:b0:d0:2a:0e:2e      Failed
00:b0:d0:2a:11:60      Successful
```

See the *OmniSwitch AOS Release 6 CLI Reference Guide* for information about the fields in this display.

See [“Configuring Host Integrity Check” on page 35-49](#) for more detailed configuration information.

Access Guardian Overview

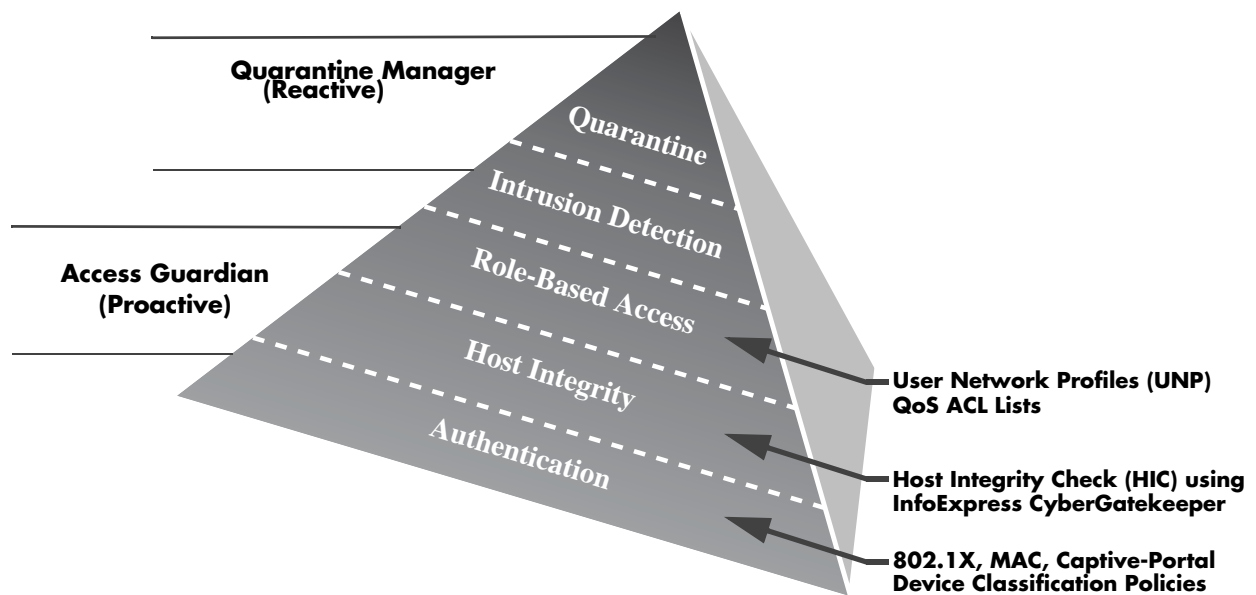
Access Guardian is a combination of authentication, device compliance, and access control functions that provide a *proactive* solution to network security. Implemented through the switch hardware and software, Access Guardian helps administrators:

- Determine who is on the network.
- Check if end users are compliant.
- Direct what end users can access within the network.

In addition to the proactive functionality of Access Guardian, the Quarantine Manager and Remediation (QMR) features provide *reactive* network security solutions. TAD and QMR help administrators:

- See what end users are doing.
- Isolate and remediate end users that are not compliant.

The Access Guardian and QMR features work together to provide a dynamic, integrated security framework. As shown in the following diagram, Access Guardian functionality provides the foundation of this framework:



The following switch-based features provide the Access Guardian functionality:

- 802.1X, MAC, and Captive Portal authentication.
- 802.1X device classification policies.
- Host Integrity Check (HIC) to verify end user device integrity.
- User Network Profiles (UNP) to classify devices, enable or disable the HIC process, and apply QoS policies to enforce device access to network resources.

This chapter documents the functionality of the Access Guardian feature.

Authentication and Classification

- **Quarantine Manager and Remediation (QMR)**—A switch-based application that interacts with OmniVista Quarantine Manager (OVQM) to restrict the network access of quarantined clients and provide a remediation path for such clients to regain their network access.

Physical devices attached to a LAN port on the switch through a point-to-point LAN connection can 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 through 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 through 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.

For non-suppliant authentication, the client MAC address is sent as username and password. The administrator can configure the password and username on the authentication server as MAC address of the client. The calling-station-ID, accounting-session-ID are also sent for authentication. All these IDs can be in uppercase or lowercase.

- **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 assign the appropriate method of authentication. Multiple authentication methods for multiple users (many users or different types of users like IP phones) 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.

For details on MAC authentication see [“Enabling MAC Authentication” on page 35-26](#).

Control Over Access Guardian Authentication (802.1x Bypass)

When a device is connected to an 802.1x port, the switch first attempts to identify and authenticate the device using EAP frames. If the device does not respond to EAP frames sent by the switch after a configurable number of attempts, then the device is identified as a non-suppliant and undergoes MAC authentication.

In some cases, however, the network administrator may want to initially apply MAC authentication to all devices (suppliant or non-suppliant) connected to the 802.1x port. In other words, the switch does not initiate 802.1x authentication; EAP frames are not sent and any received are ignored.

The advantage to applying MAC authentication first is that the MAC address of the device is initially verified (for example, checked against a RADIUS black list). Based on the outcome of the MAC authentication, the user device is then classified accordingly or can undergo subsequent 802.1x authentication.

To enforce MAC authentication as the initial authentication method for all devices connected to the 802.1x port, an 802.1x bypass operation is provided. For information about how to enable and configure 802.1x bypass options, see [“Configuring 802.1x Authentication Bypass” on page 35-40](#) for more information.

Captive Portal Bypass

Captive portal pass-through is performed globally on a bridge OmniSwitch that does not have an IP address to reach the AAA server during RADIUS Server configuration. No 802.1x configurations must be present on the bridge ports when captive-portal pass-through is configured. For enabling or disabling captive portal passthrough globally on a switch, use the **captive-portal pass-through** command with **enable** or **disable** options. When enable option is configured, the packets with Captive Portal IP address as destination are forwarded to the Layer 3 switch.

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.
- Apply a User Network Profile (UNP) to the device.
- Use Group Mobility to dynamically assign a device to a VLAN. VLAN rules are used by Group Mobility to classify user devices.
- Perform a Host Integrity Check (HIC) to determine if the end user device is compliant with network access requirements. For example, is the device using a specific version of anti-virus software. HIC is enabled or disabled through a User Network Profile.
- Apply a list of QoS policy rules to end user device traffic. A QoS policy list is associated with a UNP and applied to all devices that are associated with that profile.
- 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.

Note. Default VLAN of the port must be different from that of the UNP VLAN. UNP Policy list is not applied with UNP classified to UNP VLAN if it is same as the default VLAN assigned to the port.

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 35-38](#).

2 Group Mobility—uses Group Mobility VLAN rules and User Network Profile (UNP) mobile rules to determine the VLAN assignment for a device. UNP rules apply a profile to any device that matches the UNP rule criteria. Note that UNP mobile rules take precedence over VLAN rules. See [“What are UNP Mobile Rules?” on page 35-23](#).

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.

6 User Network Profile (UNP)—applies a pre-configured profile to a user device. The profile specifies a required VLAN ID, the optional Host Integrity Check (HIC) status, and an optional QoS policy list name. See [“User Network Profiles \(Role-Based Access\)” on page 35-22](#).

7 mac-authentication—allows basic network access to trusted devices that failed in 802.1x supplicant authentication by subjecting the user through non-supplicant MAC authentication.

It is possible to configure one or more of the classification 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:

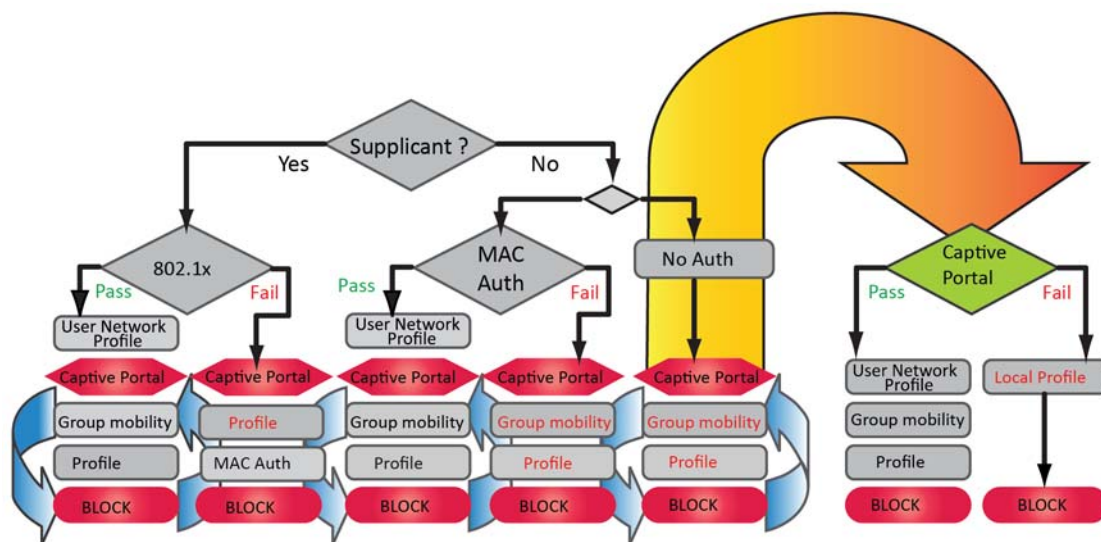
- 1** MAC-based authentication is performed.
- 2** If authentication was successful and provided a VLAN ID, the client is assigned to that VLAN and no further policy options are applied.
- 3** If a VLAN ID was not provided or authentication failed, then Group Mobility applies VLAN rules.
- 4** If there 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.

Note. Default VLAN of the port must be different from that of the UNP VLAN. UNP Policy list is not applied with UNP classified to UNP VLAN if it is same as the default VLAN assigned to the port.

See [“Configuring Access Guardian Policies” on page 35-31](#) 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 37, “Configuring 802.1X,”](#) for more information). When bypassing, 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, but not on 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 Access Guardian Policy Flow 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 through standard web browser software.

For more information, see [“Configuring Access Guardian Policies” on page 35-31](#) and [“Configuring Captive Portal Authentication” on page 35-42](#).

Host Integrity Check (End-User Compliance)

Host Integrity Check (HIC) is a mechanism for verifying the compliance of an end user device when it connects to the switch. Configurable HIC policies are used to specify, evaluate, and enforce network access requirements for the host. For example, is the host running a required version of a specific operating system or anti-virus software up to date.

The Access Guardian implementation of HIC is an integrated solution consisting of switch-based functionality, the InfoExpress compliance agent (desktop or Web-based) for the host device, and interaction with the InfoExpress CyberGatekeeper server and Policy Manager.

The switch-based functionality is provided through the configuration of a User Network Profile (UNP), which contains a configurable HIC attribute. HIC is either enabled or disabled for the profile. A UNP is a configurable option for Access Guardian device classification policies. See [“User Network Profiles \(Role-Based Access\)”](#) on page 35-22 for more information.

In addition to configuring the UNP, the HIC feature requires the configuration of global HIC parameters to enable the feature for the switch, identify the HIC server, and specify a server exception list. The HIC exception list identifies servers, such as the Web-based agent download server or a remediation server, that the host device is allowed access to during the verification process.

The InfoExpress compliance agents are used by the host device to interact with the CyberGatekeeper server. The desktop agent is installed on the device. If the desktop agent is not installed, then the switch redirects the user Web browser to a download server to obtain the Web-based agent.

The CyberGatekeeper server is configured with information that defines the criteria a host device must have installed to achieve compliance with network access requirements. The InfoExpress Policy Manager is used to define such criteria. Additional servers are configured to provide the Web-based agent and any remediation functions required to update the end user device.

Note. The HIC feature is not available unless the feature is enabled for the switch. This is true even if HIC servers are configured for the switch or the HIC attribute is enabled for a profile. See [“Configuring Host Integrity Check”](#) on page 35-49 for more information.

How it Works

The Access Guardian HIC process is triggered when a device initially connects to an 802.1X port and a device classification policy for that port applies a HIC-enabled UNP to the device. The host device is then granted limited access to the network; only DHCP, DNS, ARP, and any IP traffic between the host and any HIC-related servers is allowed. During this time, the host invokes the HIC compliance agent (desktop or Web-based) to complete the verification process.

If the HIC server determines the host is compliant, the host is then granted the appropriate access to the network. If the HIC server determines the host is not compliant, the host network access remains restricted to the HIC-related servers and any other remediation servers that can provide the host with the necessary updates to achieve compliance.

This integrated solution to provide device integrity verification is also "always-on". The HIC agent continues to check the integrity of the host device as long as the device remains connected to the switch. If the compliance agent detects a violation of the security policies or the agent itself is disabled or terminated, the HIC server notifies the switch to limit the network access for that device.

HIC Server Redundancy and Failure Mode

HIC Server Redundancy allows for Primary and Backup HIC servers to be configured. By default all HIC requests are processed by the Primary HIC server. However, if the Primary server becomes unavailable, the switch sends HIC requests to the backup server.

In case both servers are not reachable, the switch operates according to the HIC Server failure mode; either Hold or Pass-through. In Hold mode users stay in the HIC HOLD and do not have network access while the servers are down. In Pass-through mode users are treated the same as a HIC SUCCESS and have network access according to their UNP.

Determining When the Primary Server is Down

- By default, the primary server is considered the active server. If the switch does not receive a HIC-UPDATE message from the primary server for 16 seconds, the switch generates a keepalive message to the server. If the switch receives a response to the keepalive within 6 seconds it considers the server still active.
- If no response is received up to 3 additional keepalive messages are sent at 6 second intervals, if a response is received the server is considered active.
- If no response is received to the keepalive messages the switch considers the server INACTIVE and the backup server now becomes ACTIVE.
- Communication takes place with the backup server in the same way as that of the primary server and all HIC communication takes place between the backup server and the switch.
- If both servers are not reachable or if only a single server is configured the switch then operates according to the HIC Server Failure mode.

Note: The keepalive steps above are the same for the backup server if it becomes the ACTIVE server.

Determining When the Primary Server is Up

- When the backup server is ACTIVE all HIC communication takes place with the backup server in the same way as that of primary server. However, the switch continues to background poll the Primary server while it is INACTIVE.
- The frequency of sending the poll packets to the primary server is determined by the background-poll-interval.
- On reception of the first response for the background poll packet from the primary server, the switch generates a random number between range 2 to 20. This random number is used as a reconnect value. All responses from the primary server are counted and compared against the reconnect random value. When the number of continuous acknowledgments received from primary server is equivalent to the reconnect value the switch assumes the primary server is ACTIVE again.

Note: The random reconnect value prevents a HIC server from being overwhelmed by HIC requests from multiple switches simultaneously once the server becomes ACTIVE again.

- Once primary server is ACTIVE again, the backup server becomes inactive.

Background Polling Interval

By default, the background poll interval frequency is set to 16 seconds but can be configured as in the example below:

```
-> aaa hic redundancy background-poll-interval 32
```

Monitoring the Servers While Down

When both the servers are down, the backup server connection is maintained as ACTIVE and the switch continues to send keepalive packets to the backup server. In addition, background polling packets continue to be sent to the primary server so that whenever any server comes up that particular server becomes the ACTIVE server.

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 three attribute values: VLAN ID, Host Integrity Check (HIC) status, and a QoS policy list name.
- **VLAN ID.** All members of the profile group are assigned to the VLAN ID specified by the profile.
- **Host Integrity Check (HIC).** Enables or disables device integrity verification for all members of the profile group. See [“Host Integrity Check \(End-User Compliance\)” on page 35-19](#) for more information.
- **QoS policy list name.** Specifies the name of an existing list of QoS policy rules. The rules within the list are applied to all members of the profile group to enforce access to network resources. Only one policy list is allowed per profile, but multiple profiles may use the same policy list. See [“Configuring QoS Policy Lists” on page 35-51](#) for more information.
- **Maximum ingress and egress bandwidth, maximum default depth:** Specifies maximum ingress bandwidth, maximum egress bandwidth, and maximum default depth on a port. See [“Port Bandwidth Through RADIUS” on page 35-52](#) for more information.

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 controls are applied to all profile users in each building, even though they belong to different VLANs.

A UNP is a configurable option of Access Guardian device classification policies. A policy may also include 802.1X, MAC, or Captive Portal (Web-based) authentication to provide more granular control of the profile.

A device classification policy offers the following two methods for deploying a UNP:

- The UNP option is configured to specify the name of a profile. When the device classification policy is applied to an end user device, the profile attributes are applied to that device.

The Group Mobility option is configured for the policy. When this option is triggered, Group Mobility examines any VLAN rules or UNP mobile rules to determine if the device traffic matches any such rules. If there is a match with a UNP rule, the profile specified in that rule is applied to the device. Note that UNP rules take precedence over VLAN rules.

User profiles and UNP mobile rules must already exist in the switch configuration before they are deployed through Access Guardian device classification policies. See [“Configuring User Network Profiles” on page 35-51](#) and [“What are UNP Mobile Rules?” on page 35-23](#) for more information.

What are UNP Mobile Rules?

Classifying devices with UNP mobile rules allows the administrator to assign users to a profile group based on the source IP or source MAC address of the device. For example, 802.1X port 1/10 is configured with a device classification policy that uses Group Mobility. Next, a UNP mobile rule is configured with 10.1.1.0 as the source IP value and “Engineering” as the user profile. Any devices connecting to port 1/10 with a source IP address that falls within the 10.1.1.0 network is assigned to the Engineering profile.

If the UNP option of a device classification policy is used to classify users into profile groups, all devices that the policy authorizes for a specific port are assigned to the profile regardless of their source IP or MAC address values. UNP rules narrow the selection of user devices for profile groups.

When the Group Mobility option of an Access Guardian device classification policy is used to deploy a UNP, Group Mobility checks to see if any UNP mobile rules (also referred to as device classification rules) exist in the switch configuration. If so, the UNP rules are applied, as they take precedence over VLAN rules. If there are no applicable UNP rules, then the VLAN rules are applied.

UNP rules differ from VLAN rules in that they assign a user profile to a device that matches the rule. The profile then determines the VLAN assignment for the device. VLAN rules directly assign a device to the VLAN for which the matching rules are configured.

There are three types of UNP mobile rules available: IP address, MAC address, and MAC address range. Each type of rule specifies the criteria that a device must match and the name of a user profile that is applied to the device when the match occurs.

For more information about UNP rules, see [“Configuring User Network Profile Mobile Rules” on page 35-54](#). For more information about Group Mobility VLAN rules, see [Chapter 9, “Defining VLAN Rules.”](#)

Dynamic UNP

The OmniSwitch can associate a client MAC address with a UNP based on an authentication result, such as 802.1X or MAC authentication, or based on classification rules, such as IP or MAC ranges.

Dynamic UNP extends this capability by enhancing the protocol between the HIC server and the OmniSwitch allowing the HIC server to return the UNP that a user must be associated with. This allows users to be classified into UNPs based on Active Directory group memberships, machine specific parameters or any other parameters the HIC agent supports. Once classified into a UNP, specific access rights can be enforced by applying the policy list associated with the UNP to the user.

CMD-RESET Keyword

In the case of HIC failure if a UNP is returned with a special keyword of **CMD-RESET**, the associated MAC address is flushed from the switch and forced to re-initiate the 802.1x classification.

Dynamic UNP Operation Summary Table

| | HIC PASS | HIC FAIL |
|--|---|--|
| Valid UNP returned and HIC enabled | Classify the client based on UNP returned from HIC server. | Classify the client based on UNP returned from HIC server. |
| Invalid UNP returned (HIC Disabled) | Client remains classified in current UNP. | Client remains classified in current UNP. |
| Unknown UNP returned | Client remains classified in current UNP. | Client remains classified in current UNP. |
| UNP not returned | Client remains classified in current UNP. | Client remains classified in current UNP. |
| UNP with keyword CMD-RESET returned | Client remains classified in current UNP. (CMD-RESET ignored) | User MAC is flushed from switch and re-initiates 802.1x process. |

Dynamic UNP Operation

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.

Quality of Service (QoS)

The Access Guardian User Network Profile (UNP) feature provides the ability to assign a list of QoS policy rules to a profile. The rules contained in the list are applied to any device that is assigned to the UNP. Consider the following guidelines when configuring policy lists for user profiles:

- QoS policy rules and policy lists are configured using the QoS switch feature. Configuration of these items is required before the list is assigned to a UNP.
- Configuring QoS policy lists is not allowed if VLAN Stacking Services or if QoS inner VLAN or inner 802.1Q tag policies are configured for the switch.
- Only one QoS policy list per UNP is allowed, but multiple profiles can use the same UNP. Up to 13 policy lists (including the default list) are allowed per switch.
- A default QoS policy list always exists in the switch configuration. Any QoS policies that are not assigned to a user profile belong to the default list, unless specified otherwise when the policy is created.
- If a QoS policy list is configured for a user profile, only the policy rules in the list are applied to traffic from devices to which the profile was applied. Any default list policy rules are not applied in this case.
- If a QoS policy list is not specified for a user profile, then any policies from the default list are applied to profile devices.
- If a policy rule is enabled, it is active for all policy lists to which it belongs. If one of the policy lists is disabled, the rule is still active for all the other lists.
- If a policy rule is disabled, it is no longer active in any policy list to which it belongs, even if the list is still enabled.

Host Integrity Check - InfoExpress

- VLAN Stacking Ethernet services are not available when the HIC feature is configured for the switch. These two features are mutually exclusive; only one of them can run on the switch at any given time.
- The Host Integrity Check (HIC) feature on the switch interacts with compliance agents and the CyberGatekeeper server from InfoExpress. The compliance products consist of a desktop and Web-based agent. Refer to the *OmniSwitch Release Notes* for information about platform and browser support for both types of agents.

Refer to the InfoExpress documentation for information about how to configure the CyberGatekeeper server and other related products.

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 an n 802.1X supplicant (or device). Optional parameters can 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 is used for authenticating 802.1X ports. If **rad1** becomes unavailable, the switch uses **rad2** for 802.1X authentication. When this command is used, 802.1X is automatically enabled on 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. 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 non-supplicant authentication and other details on configuring authentication servers, see chapter [Chapter 35, “Configuring Access Guardian”](#)

For more information about using MAC authentication and classifying non-suppliant devices, see [“Authentication and Classification” on page 35-15](#), [“Configuring Access Guardian Policies” on page 35-31](#), and [“Configuring User Network Profiles” on page 35-51](#).

MAC accounting

Use the **aaa accounting mac** command to create an accounting server entry for the non-suppliant mac-based authentication. This verifies if the RADIUS server is configured as the authentication server for MAC.

The following example specifies the accounting server *rad1* for the non-suppliant MAC-based authentication:

```
-> aaa accounting mac rad1 local
```

Enabling an Authentication Server Down Policy

An authentication server down policy is used to classify devices attempting to authenticate through 802.1x 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 35-51](#) 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, re-authentication of the device is tried after a specific 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. When device authentication fails due to an unreachable RADIUS server, an event message is sent to the switch logging utility (swlog). See [Chapter 44, “Using Switch Logging,”](#) for more information.

Critical Voice VLAN

The Critical Voice VLAN feature allows classifying and differentiating the IP phone traffic from the data traffic when the authentication server is down.

In the presence of an authentication server, the IP phone MAC address is authenticated against the authentication server and the traffic is classified into a voice domain VLAN that is returned from the RADIUS server.

When the authentication server is down, the User Network Profile (UNP) must be configured to classify packets. The user-network-profile is associated to the voice policy and the UNP is applied to the detected IP phone traffic using the critical voice VLAN.

show aaa server command displays the reachability status of different RADIUS servers configured on the switch.

MAC Verification to Classify IP Phone Traffic

When the RADIUS server is down, the MAC address is first classified as IP phone traffic or non-IP Phone traffic. The MAC address is verified against the Link Layer Discovery Protocol (LLDP) database for classification.

The LLDP signature and Type-Length-Value (TLV) are used to determine if the MAC address is an IP phone.

The following LLDP signatures are used:

- System is a “Telephone”
- System is a LLDP-MED Endpoint Class III (end user communication appliances supporting IP media)
- System is a LLDP-MED endpoint supporting LLDP-MED Network Policy and Extended Power through MDI

The following TLVs are verified:

- 802.1AB System Capabilities TLV
- The LLDP-MED Capabilities TLV

On identifying the type of traffic, the following actions are performed based on the configuration:

- If it is IP phone traffic, then the voice-policy UNP is applied.
- If it is non-IP phone traffic, then the default UNP is applied.
- If no UNP is configured, then the MAC is blocked.

Classification Guidelines:

1 If the authentication server is down when an IP phone is connected for the first time and the initial packet from the IP phone is not an LLDP packet, such as a EAP packet, the IP phone will not be seen as an IP phone and will not be classified according to the voice-policy UNP. In such case, the IP phone will be classified in data-auth-server-down policy.

2 After being classified, if the IP phone then sends an LLDP packet, the information will be passed on to check if that device is already classified based on data-auth server down policy. If yes, then the client will be re-authenticated automatically (does not wait for the re-auth interval) and moved to the voice-auth server down policy, if the server is still unreachable.

3 During this re-authentication process, the device/authentication details in supplicant or non-suppliant table will be modified to initial state, but the old VPA association and the MAC address will not be deleted/modified until the server returns the result of the new authentication. If the returned/classified VLAN/UNP of new authentication is same as the old VLAN/UNP, then the device/authentication details in supplicant or non-suppliant table will be updated. If the returned/classified VLAN/UNP of new authentication is different from the old VLAN/UNP, then the VPA is updated for new VLAN. Additionally, the old MAC entry for that device will be removed and new entry will be added.

4 The above re-authentication procedure is applicable for the re-authentication scenarios listed below:

- Auth-server down re-authentication (auth-server down re-authentication only applies to data devices).
- 802.1x re-authentication initiated by the switch.
- Client initiated EAP start.

5 During the re-authentication, the 'port state' in [show 802.1x users](#) for that client shall display as 'Authenticating' until the authentication gets completed. Meanwhile, MAC address table shall have the client in previously classified VLAN. The MAC entry may get changed based on the result of the re-authentication.

Configuring Critical Voice VLAN

Critical Voice VLAN is configured along the auth-server-down functionality. A voice policy is associated to the UNP to classify the IP phone traffic. To configure the critical voice VLAN, use the [802.1x auth-server-down policy](#) command.

In the following example, a user network profile "voice" is configured and associated to the voice VLAN 10. In order to authenticate the IP phone traffic for the UNP "voice" when the authentication server is down, it is configured for voice policy classification using the critical voice VLAN.

```
-> aaa user-network-profile name "voice" vlan 10
-> 802.1x auth-server-down voice-policy user-network-profile voice
```

Note. This feature operates only with POE IP phones.

Removing the Critical Voice VLAN Configuration

To remove the critical voice VLAN or the voice policy from the UNP, use the no form of the [802.1x auth-server-down policy](#) command. For example:

```
-> 802.1x auth-server-down no voice-policy user-network-profile voice
```

On removal of the voice policy, the UNP is applied with the auth-server-down policy.

Verifying the Critical Voice VLAN Configuration

To verify the critical voice VLAN configuration, use the [show 802.1x auth-server-down](#) command. Example:

```
-> show 802.1x auth-server-down
Status = Enabled
Re-authentication Interval = 30 seconds
Delay-Learning Period = 120 seconds
Classification policy = UNP 'unp1', block
Classification Voice Policy = UNP 'unp2'
```

To verify the classification type, use the **show 802.1x non-suppliant** command. For example:

| Slot | MAC | MAC Authent | Classification | Vlan | User |
|-------|-------------------|------------------|-------------------|---------|--------------|
| Port | Address | Status | Policy | Learned | Name |
| 01/10 | 00:80:9f:6b:dc:1b | Auth Svr Timeout | AuthSvrDwn-VP-UNP | 1 | 000000002c83 |

The IP Phone when classified on a Voice Policy shows the classification policy type as AuthSvrDwn-VP-UNP.

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 is set up with defaults listed in “802.1X Defaults” of the [Chapter 37, “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 7, “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 can retransmit an authentication request to the user.

If it is necessary to change the default values of these parameters, see [Chapter 37, “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 35-17](#) 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 37, “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 35-17](#), 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 (for example, 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.
- Configuring a policy to apply a User Network Profile (UNP) requires the name of an existing profile. In addition, certain profile attributes may also require additional configuration. See [“Configuring User Network Profiles” on page 35-51](#) for more information.

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. 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 through 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 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
user-network-profile
vlan
default-vlan
block
captive-portal
mac-authentication
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 supplicant policy 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> |
| 802.1x 2/6 supplicant policy authentication fail mac-authentication | <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 2/6.</p> <p>If the device fails 802.1x authentication, then the basic network access is provided to trusted devices that failed in 802.1x supplicant authentication by subjecting the user through non-supplicant MAC authentication.</p> <p>After authentication, the users get classified based on the returned VLAN or based on local authorization on non supplicant policy.</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 performs 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
user-network-profile
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 non-suppliant policy 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 suppliant devices to VLANs that are *not* authenticated VLANs.

To configure a non-suppliant policy that does *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
user-network-profile
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 suppliant 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> |
| 802.1x 2/1 non-suppliant 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. |

| Supplicant Policy Command Example | Description |
|--|--|
| 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 from 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 35-32 and [“Configuring Non-supplicant Policies”](#) on page 35-34 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 captive-portal policy 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 802.1x Authentication Bypass

The authentication method determines if the client device is classified as a supplicant (802.1x-enabled) device or a non-suppliant (non-802.1x) device. This in turn triggers Access Guardian to apply either supplicant or non-suppliant device classification policies to the client device. See [“Configuring Access Guardian Policies” on page 35-31](#) for more information.

By default, the switch initially sends EAP frames to a client device to determine whether 802.1x authentication is applied to the device. If the client does not qualify for 802.1x authentication (does not respond to EAP frames), MAC authentication is used.

An 802.1x bypass operation is provided to specify that Access Guardian must apply MAC authentication first to any device (suppliant or non-suppliant) connected to the 802.1x port. In addition, the bypass operation provides configurable options that are used to specify if subsequent 802.1x authentication is performed on the device based on the results of MAC authentication.

Configuring 802.1x authentication bypass is done using the [802.1x supplicant bypass](#) and [802.1x non-suppliant allow-eap](#) commands. The [802.1x supplicant bypass](#) command enables or disables the bypass operation. The following [802.1x non-suppliant allow-eap](#) command parameters determine if subsequent 802.1x authentication is attempted on the device after MAC authentication:

- **pass**—802.1x authentication is attempted if the device passes the initial MAC authentication. If the device fails MAC authentication, 802.1x authentication is bypassed (EAP frames are ignored) and the device is classified as a non-suppliant.
- **fail**—802.1x authentication is attempted if the device fails the initial MAC authentication. If the device passes MAC authentication, 802.1x authentication is bypassed (EAP frames are ignored) and the device is classified as a non-suppliant.
- **noauth**—802.1x authentication is automatically attempted as there is no MAC authentication available for this port.
- **none**—802.1x authentication is permanently bypassed. Only MAC authentication is performed and the device is classified as a non-suppliant.

Configuration Guidelines

Consider the following guidelines before configuring 802.1x authentication bypass:

- The 802.1x bypass operation is only supported on 802.1x ports configured for auto access control mode. See [“Enabling 802.1X on Ports” on page 35-30](#) for more information about configuring the access control mode.
- If a port has supplicants connected, and 802.1x bypass is enabled for that port, the supplicants are automatically logged off to undergo authentication according to the enabled bypass configuration.
- When the 802.1x bypass configuration is modified or disabled, any non-suppliant devices are automatically logged off the port. This will free up those devices to undergo the authentication specified by the new bypass configuration.
- If re-authentication is configured for the 802.1x port and supplicant bypass is enabled, the MAC authentication followed by 802.1x authentication is initially performed as configured. However, only 802.1x authentication is performed during the re-authentication process, so there is no recheck to see if the MAC address of the user device is restricted.

- When successful MAC authentication returns a VLAN ID or User Network Profile (UNP) and the 802.1x bypass operation is configured to initiate 802.1x authentication when a device passes MAC authentication, the device is *not* moved into that VLAN or UNP. Instead, the device is moved into the VLAN or UNP returned by 802.1x authentication. If 802.1x authentication does not provide such information, the device is moved based on the supplicant device classification policies for the port.
- When supplicant bypass is enabled after MAC authentication, till it completes the supplicant authentication, the port will be in `mac_authenticated_await8021x` state.
- Configuring 802.1x supplicant bypass is not allowed on ports where the 802.1x supplicant polling retry count is set to zero. Both operations are mutually exclusive on the same port.
- Using the **802.1x non-supplicant allow-eap** command with the **none** parameter is similar to setting the supplicant polling retry counter to zero (see “[Configuring the Number of Polling Retries](#)” section in [Chapter 34, “Configuring 802.1X,”](#)). However, the functionality configured with each command differs as follows:
 - > When the supplicant polling retry is set to zero, EAP frames are ignored. MAC authentication is only triggered when a non-EAP frame is received, which is when the supplicant times out and is in an open state.
 - > When the allow EAP is set to none, EAP frames are ignored but MAC authentication is triggered when the first EAP frame is received and the supplicant is not in an open state.

Example: Supplicant Bypass with allow-eap as Fail

The following CLI command configures 802.1x bypass on port 2/1 and specifies the non-supplicant fail branch that triggers 802.1x authentication if the initial MAC authentication fails.

```
-> 802.1x 2/1 supplicant bypass enable
-> 802.1x 2/1 non-supplicant allow-eap fail
```

The resulting Access Guardian authentication process for a device connected to 802.1x port 2/1 is as follows for this example:

- MAC authentication is triggered when the first frame from the new user is received, whether it is an EAP frame or not.
- EAP frames for this user are ignored until MAC authentication completes (RADIUS returns an Access-Accept or a Access-Reject response).
- Once the initial MAC authentication passes (that is, Access-Accept), 802.1x authentication is bypassed for this user and all EAP frames are ignored. The user is permanently authenticated through MAC authentication and 802.1x is permanently bypassed.
- Once the initial MAC authentication fails (that is, Access-Reject), 802.1x authentication is allowed for this user. The user is authenticated through 802.1x authentication. During this transition, the EAP frames are allowed and the switch must force the supplicant to restart a fresh EAP session by sending a multicast Request Identity EAPOL on the port. This is because the supplicant may have already sent an EAPOL Start.

Configuring Captive Portal Authentication

Captive Portal authentication allows Web browser clients to authenticate through the switch using 802.1x or MAC authentication through 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.

- **Ensure that 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 35-32 and “[Configuring Non-supplicant Policies](#)” on page 35-34.
- **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 35-38.
- **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 35-43.
- **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 35-43.

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 log in 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 35-7.

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 Web-based login page components, create one or more of the following file types:

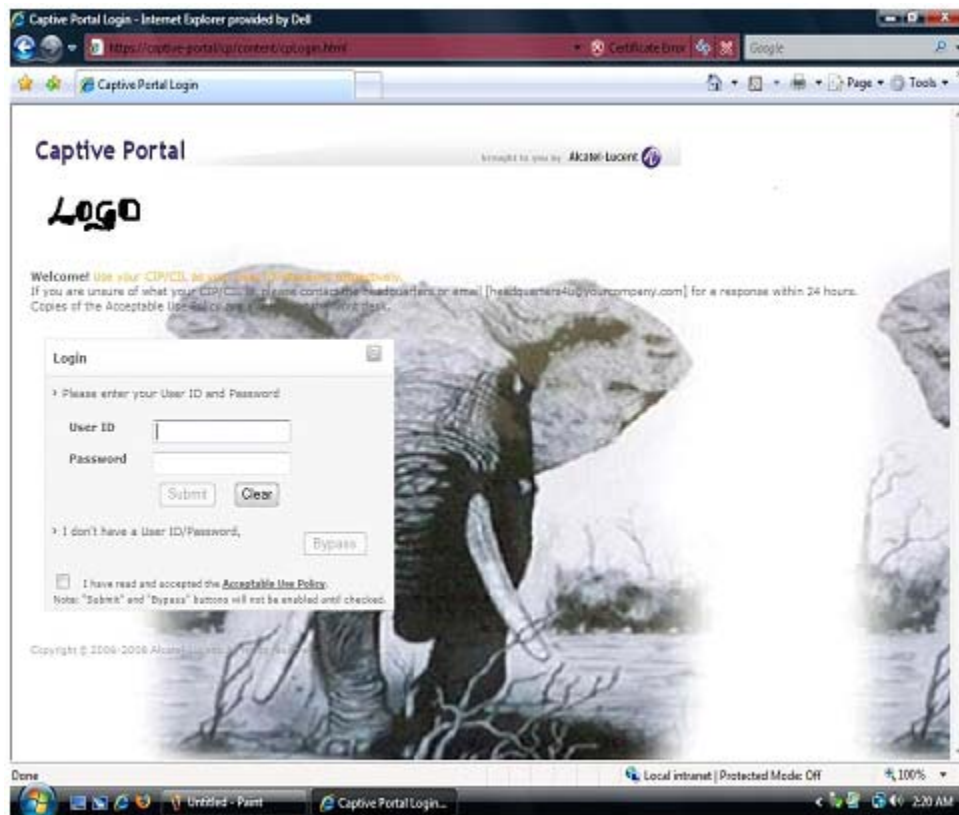
- **logo.gif, logo.jpg, or logo.png**—Use these files to provide a company logo that Captive Portal displays on all pages.
- **background.gif, background.jpg, or background.png**—Use these files to provide a page background image that Captive Portal displays on all pages.
- **cpPolicy.html** —The User 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, ensure that the filename matches the filename exactly as shown in the list of file types described.
- 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 **.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 authentication conditions are met, Access Guardian places the device MAC address in a Captive Portal state. This means that the switch does not learn the device MAC address and a Web browser session is required to proceed with the authentication process.

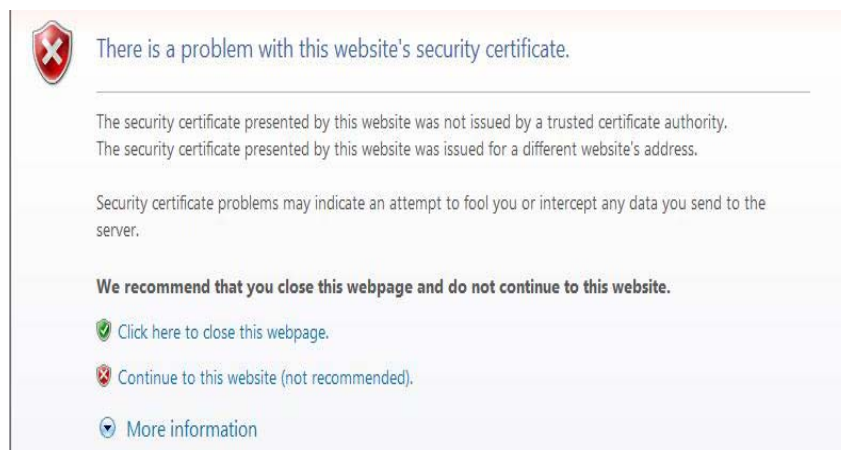
Note. Captive Portal does not require the configuration of IP interfaces, a UDP Relay agent, or an external DHCP server to provide an IP address for the client device. A temporary IP address derived from the Captive Portal subnet is assigned to the client for use during the authentication process. For more information, see [“Configuring Captive Portal Authentication” on page 35-42](#).

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 attempts 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 35-42](#) for more information).

A certificate warning message can 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:



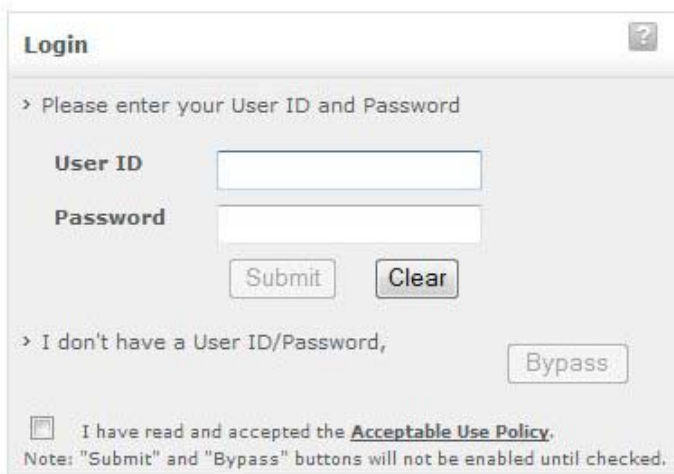
When the browser window opens and after the certificate warning message, if any, is cleared, Captive Portal displays a login screen similar to the one shown in the following example:

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

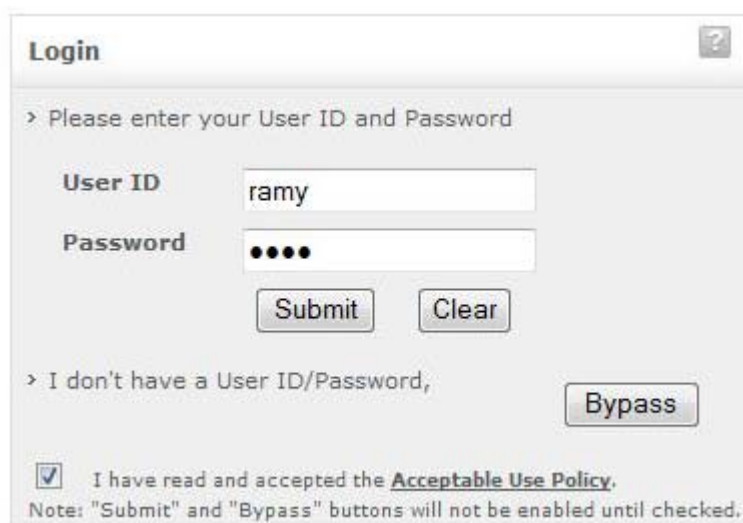
> I don't have a User ID/Password,

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 follows:



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 35-47). 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 in to 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” and note 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 35-48 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.

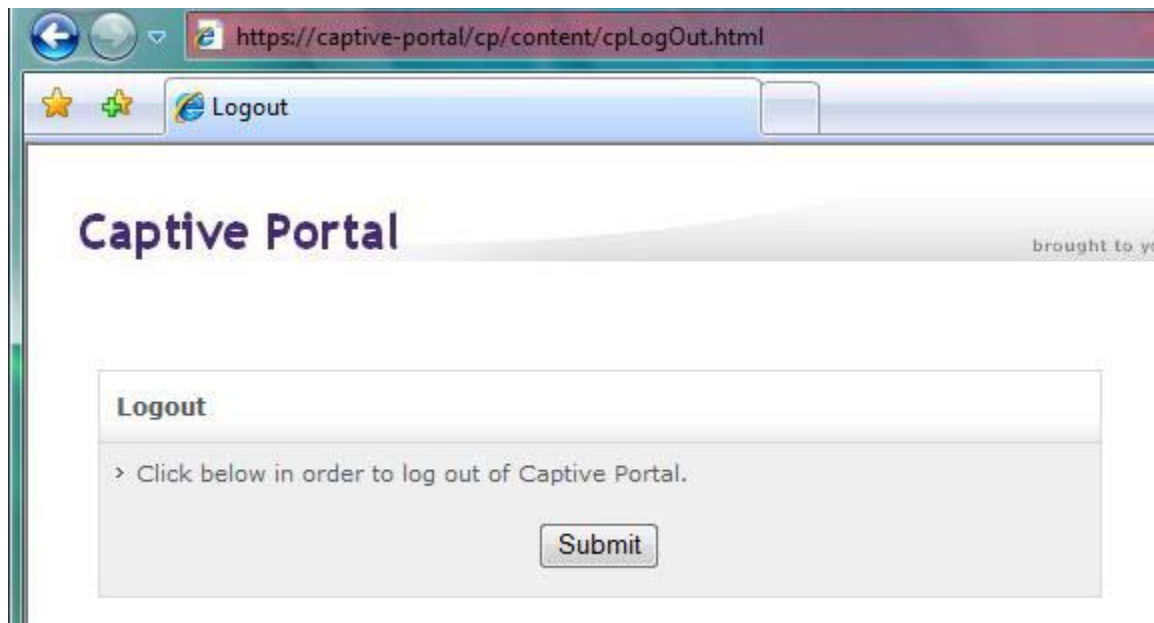
Bypassing Captive Portal Login

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. 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 35-38.

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 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 35-43.

Configuring Host Integrity Check

The Access Guardian Host Integrity Check (HIC) feature provides an integrated solution for device integrity verification. This solution involves switch-based functionality that interacts with the InfoExpress HIC server (CyberGatekeeper) and host devices using InfoExpress compliance agents.

This section describes how to configure the switch-based functionality. See the InfoExpress user documentation for more information regarding the configuration of compliance agents and the CyberGatekeeper server.

The Host Integrity Check (HIC) process is triggered when a HIC-enabled User Network Profile (UNP) is applied to a client device. See [“User Network Profiles \(Role-Based Access\)” on page 35-22](#) for more information. When a profile is created, HIC is disabled by default. To enable HIC for the profile, use the [aaa user-network-profile](#) command. For example:

```
-> aaa user-network-profile name Engineering vlan 500 hic enable
```

In addition to enabling HIC for a UNP, the following configuration tasks are involved in setting up the HIC feature to run on the switch:

1 Configure the identity of the HIC server. Use the [aaa hic server-name](#) command to configure the name and IP address of the InfoExpress CyberGatekeeper server, a shared secret, and the UDP port number used for HIC requests.

```
-> aaa hic server-name hic-srv1 ip-address 2.2.2.2 key wwtoe
```

Note that configuring the server is required before HIC can be enabled for the switch.

An option **prompt-key** is provided which can be used to enter the shared key in a masked format rather than as clear text. When this option is used, a prompt appears prompting to enter the shared key. Key needs to be re-entered, and only if both the entries match, command is accepted. Key provided in this mode is not displayed on the CLI as text.

For example,

```
-> aaa hic server-name hic1 ip-address 1.1.1.1 prompt-key
Enter Key:  *****
Re-enter Key:  *****
```

2 Configure the Web agent download server URL. A host can use the InfoExpress desktop compliance agent or a Web-based agent. If the desktop agent is not installed on the host, the switch redirects the host to a Web agent download server. The URL of the download server is configured for the switch using the [aaa hic web-agent-url](#) command.

```
-> aaa hic web-agent-url http://10.10.10.10:2146
```

When the HIC process is initiated for a host device, the host has limited access to the network for communicating with the HIC server and any servers included in the exception list. Make sure the Web agent download server is added to the server exception list, as described below.

3 Configure a server exception list. There are specific servers that a host device may need access to during the HIC process. For example, if the host is going to use the Web-based compliance agent, access to the Web agent download server is required. Use the [aaa hic allowed-name](#) command to add the name and IP address of up to four servers to the HIC server exception list.

```
-> aaa hic allowed-name webserv1 ip-address 123.10.5.1 ip-mask 255.255.255.0
```

4 Configure a custom proxy port number. By default, the switch uses 8080 for the host proxy port number. If a different number is used by the host device, use the **aaa hic custom-proxy-port** command to configure the switch to use the host value.

```
-> aaa hic custom-proxy-port 8878
```

5 Enable the HIC feature for the switch. By default, the HIC feature is disabled for the switch. This means that all HIC functionality is disabled. For example, if the HIC attribute of a UNP is enabled, the HIC process is not invoked when the profile is applied if the HIC feature is not enabled for the switch. Use the **aaa hic** command to enable or disable the HIC feature for the switch.

```
-> aaa hic enable
```

Note that enabling the HIC feature for the switch is not allowed if the HIC server information is not configured. Check to see if the server configuration exists before attempting to enable this feature.

Use the **show aaa hic host** command to see a list of host MAC addresses the switch has learned and the HIC status for each host. The **show aaa classification-rule**, **show aaa classification-rule**, and **show aaa hic server-failure policy** commands provide information about the HIC status and configuration for the switch.

For more information about HIC, see “[Host Integrity Check \(End-User Compliance\)](#)” on page 35-19.

Configuring HIC Redundancy

The role of the servers can be either primary or backup which is specified when the servers are configured. Only one server per role is allowed and a backup server can only be configured if the primary server exists. For example, to configure both a Primary and Backup server enter the following:

```
-> aaa hic server-name hic-srv1 ip-address 2.2.2.2 key secret1 role primary
-> aaa hic server-name hic-srv2 ip-address 2.2.2.22 key secret2 role backup
```

HIC Server Failure Mode

In case both servers are unavailable the HIC Server Failure Mode can be used to determine how users must be handled while the servers are unavailable.

- **Hold Mode:** This is default mode. Hosts stay in their UNP and in a HIC HOLD state. Users in this state are treated the same as a HIC FAILED and do not have network access:

```
-> aaa hic server-failure mode hold
```

- **Pass-Through Mode (Fail Open):** In Pass-through mode HIC users are moved to the HIC PASSTHROUGH state. Users in this state are treated the same as a HIC SUCCESS and have network access according to the policy list for their UNP, for example:

```
-> aaa hic server-failure mode passthrough
```

- **Mapping Users to Temporary UNP:** Mapping can be used to move all users in the HIC IN PROGRESS state from their current UNP to a temporary UNP while the servers are down, for example:

```
-> aaa hic server-failure policy user-network-profile change unpx to unpy
```

While the servers are down, all HIC new and existing HIC users in the HIC IN PROGRESS state are temporarily moved to the unpy. When any one of the HIC servers comes back up the HIC hosts in unpy is moved back to unpx and restart the HIC validation.

Configuring User Network Profiles

User Network Profiles (UNP) are applied to host devices using Access Guardian device classification policies. However, configuring the profile name and the following associated attributes is required prior to assigning the profile using device classification policies:

- **VLAN ID.** All members of the profile group are assigned to the VLAN ID specified by the profile.
- **Host Integrity Check (HIC).** Enables or disables device integrity verification for all members of the profile group. See [“Host Integrity Check \(End-User Compliance\)” on page 35-19](#).
- **QoS policy list name.** Specifies the name of an existing list of QoS policy rules. The rules within the list are applied to all members of the profile group. Only one policy list is allowed per profile, but multiple profiles may use the same policy list.

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, enable HIC, and apply the rules from the “temp_rules” policy list:

```
-> aaa user-network-profile name guest_user vlan 500 hic enable policy-list-name temp_rules
```

- **Maximum ingress and egress bandwidth, maximum default depth:** Specifies maximum ingress and egress bandwidth limiting, and maximum default depth configuration on a port. See [“Port Bandwidth Through RADIUS” on page 35-52](#) for more information.

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 35-22](#).

Configuring QoS Policy Lists

One of the attributes of a User Network Profile (UNP) specifies the name of a list of QoS policy rules. This list is applied to a user device when the device is assigned to the user profile. Using policy lists allows the administrator to associate a group of users to a set of QoS policy rules.

Configuring the QoS list is required prior to associating the list with a UNP. In addition, the policy rules must exist before they are assigned to a policy list.

The **policy list** command is used to group a set of QoS policy rules into a list. For example, the following commands create two policy rules and associates these rules with the “temp_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 temp_rules type unp rules r1 r2
-> qos apply
```

Note the following guidelines when configuring QoS policy rules and lists:

- A default policy list exists in the switch configuration. Rules are added to this list when the rule is created. A rule can belong to multiple policy lists. As a result, the rule remains a member a of the default list even when it is subsequently assigned to additional lists.

- Each time a rule is assigned to a policy list, an instance of that rule is created. Each instance is allocated system resources. 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
```

- Up to 13 policy lists (including the default list) are supported per switch. Only one policy list per UNP is allowed, but a policy list can be associated with multiple profiles.
- 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.

Use the **show policy list** command to display the QoS policy rule configuration for the switch.

Port Bandwidth Through RADIUS

This feature applies maximum ingress and egress bandwidth limiting, maximum default depth on a port on the basis of UNP classification. When a user is successfully authenticated under a UNP policy either through RADIUS returned UNP attribute or through a local UNP policy, bandwidth limitations are applied on the port.

Configuring Bandwidth Profiling on a UNP

User can be a supplicant, non-supplicant, or a captive portal client. Bandwidth profiling per user is not supported. If multiple users are authenticated on a port, then bandwidth limitation of the latest user overwrites the existing bandwidth limitations, if any.

Use the following CLI command to configure bandwidth limitation:

```
-> aaa user-network-profile name "profile1" vlan 50 maximum-ingress-bandwidth  
1024 maximum-egress-bandwidth 256 maximum-default-depth 128
```

Note.

- Bandwidth limitation applied on a port by UNP classification is not removed on user log out or aging. User can either overwrite the bandwidth through “QoS port” command or disable 802.1x on the port to remove the UNP applied bandwidth active on the port. Administratively bringing down the port also removes the applied UNP configurations on the port.
- Run time modification of ingress and egress bandwidth is allowed but it does not affect the user already authenticated, instead, it is applied on the new user authenticating under UNP. That is, if a user modifies the UNP profile during the run time, the modified profile is not applied to those users already authenticated with the same UNP. If any user disconnects and then authenticates again, then the modified profile is applied to the user.

- If any parameter (egress bandwidth, ingress bandwidth, default depth) associated with a UNP profile is modified, then the other parameters must be reconfigured, otherwise, they will be set to their default values.

For example, consider a UNP profile with egress and ingress bandwidth as 100M, and default depth as 10M. Modify only the egress bandwidth as 200M. In this scenario, only the modified egress bandwidth is considered, but the ingress and the default depth is set to their default values unless specifically configured to the required values.

Multiple User Authentication on the Same Port

If multiple users are authenticated on the same 802.1x port and get classified on either RADIUS returned attributes or through locally configured authentication policies, then the bandwidth associated to the latest authenticated user overwrites the previous bandwidth associated. If there is no bandwidth associated to the new user, then no rate limitations are enforced, and previously set bandwidth is applied to the new user authenticated.

There is no priority among bandwidth profile provided by "QoS port" or "UNP". Any latest change overwrites the previous bandwidth limitation applied on the port. Some scenarios are stated below.

| Bandwidth Profile | Action |
|---|--|
| If a user authenticates in a UNP with no UNP bandwidth profile (that is, no ingress and no egress bandwidth configured) | The port ingress and egress bandwidth currently set on the port is considered. |
| If a user authenticates in a UNP with a bandwidth profile with only ingress bandwidth | The port ingress bandwidth is overwritten by the UNP ingress bandwidth. However, the port egress bandwidth configuration is not changed. |
| If a user authenticates in a UNP with a bandwidth profile with only egress bandwidth | Only the port egress bandwidth is overwritten by the UNP egress bandwidth. However, the port ingress bandwidth configuration is not changed. |
| If a user authenticates in a UNP with a bandwidth profile with both ingress and egress bandwidth | The port ingress and egress bandwidth are overwritten by the UNP ingress and egress bandwidth. |

Note. The same bandwidth behavior applies when the user is authenticated with QoS port bandwidth, QoS port configuration being the latest configuration.

Configuring User Network Profile Mobile Rules

The Group Mobility device classification policy option uses both VLAN mobile rules and UNP mobile rules to classify user devices. VLAN rules dynamically assign users into VLANs. UNP rules specify a user profile that is applied to the user device. The profile determines the VLAN assignment for the device.

Note that UNP mobile rules take precedence over VLAN rules. For information about how to configure VLAN rules, see [Chapter 9, “Defining VLAN Rules.”](#) For more information about user profiles, see [“Configuring User Network Profiles” on page 35-51.](#)

There are three types of UNP mobile rules available: MAC address, MAC address range, and IP network address rules. To configure a UNP MAC address rule, use the **aaa classification-rule mac-address** command. For example, the following command applies the “accounting” profile to a device with the specified source MAC address:

```
-> aaa classification-rule mac-address 00:00:2a:33:44:01 user-network-profile
name accounting
```

To configure a UNP MAC address range rule, use the **aaa classification-rule mac-address-range** command. For example, the following command applies the “accounting” profile to a device with a source MAC address that falls within the specified range of MAC addresses:

```
-> aaa classification-rule mac-address-range 00:00:2a:33:44:01 00:00:2a:33:44:10
user-network-profile name accounting
```

To configure a UNP IP address rule, use the **aaa classification-rule ip-address** command. For example, the following command applies the “accounting” profile to a device with the specified source IP address:

```
-> aaa classification-rule ip-address 10.1.1.1 user-network-profile name
accounting
```

Use the **show aaa classification-rule** command to verify the UNP mobile rule configuration for the switch. For more information about UNP rules, see [“What are UNP Mobile Rules?” on page 35-23.](#)

Configuring Dynamic UNP

Initially an 802.1x client has to be classified based on a default UNP policy with HIC enabled for that UNP. When the HIC operation is performed for a user the HIC server must be configured to return a UNP name in the policy update packet by associating a UNP name to a specific MAC address. Upon receiving the policy update packet from the HIC server the OmniSwitch dynamically moves the HIC user from the current UNP to the one returned by the server.

The example below configures a default UNP named ‘default_unp’ where both supplicant and non-supplicant users are classified. The HIC server must be configured to return a UNP named ‘dynamic_unp’ based on the user MAC or other parameters:

```
-> aaa user-network-profile name default_unp vlan 10 hic enable
-> 802.1x 2/23 non-supplicant policy user-network-profile default_unp fail block
-> 802.1x 2/24 supplicant policy user-network-profile default_unp block
-> aaa user-network-profile name dynamic_unp vlan 20 hic enable
```


OmniAccess Stellar AP Integration

Access Guardian provides the framework through which OmniAccess Stellar Access Points (APs) connected to an OmniSwitch are detected, learned, and managed. Wireless client traffic is then forwarded from the AP device to the OmniSwitch and onto the wired network. This integration provides a unified wireless over wired network access solution.

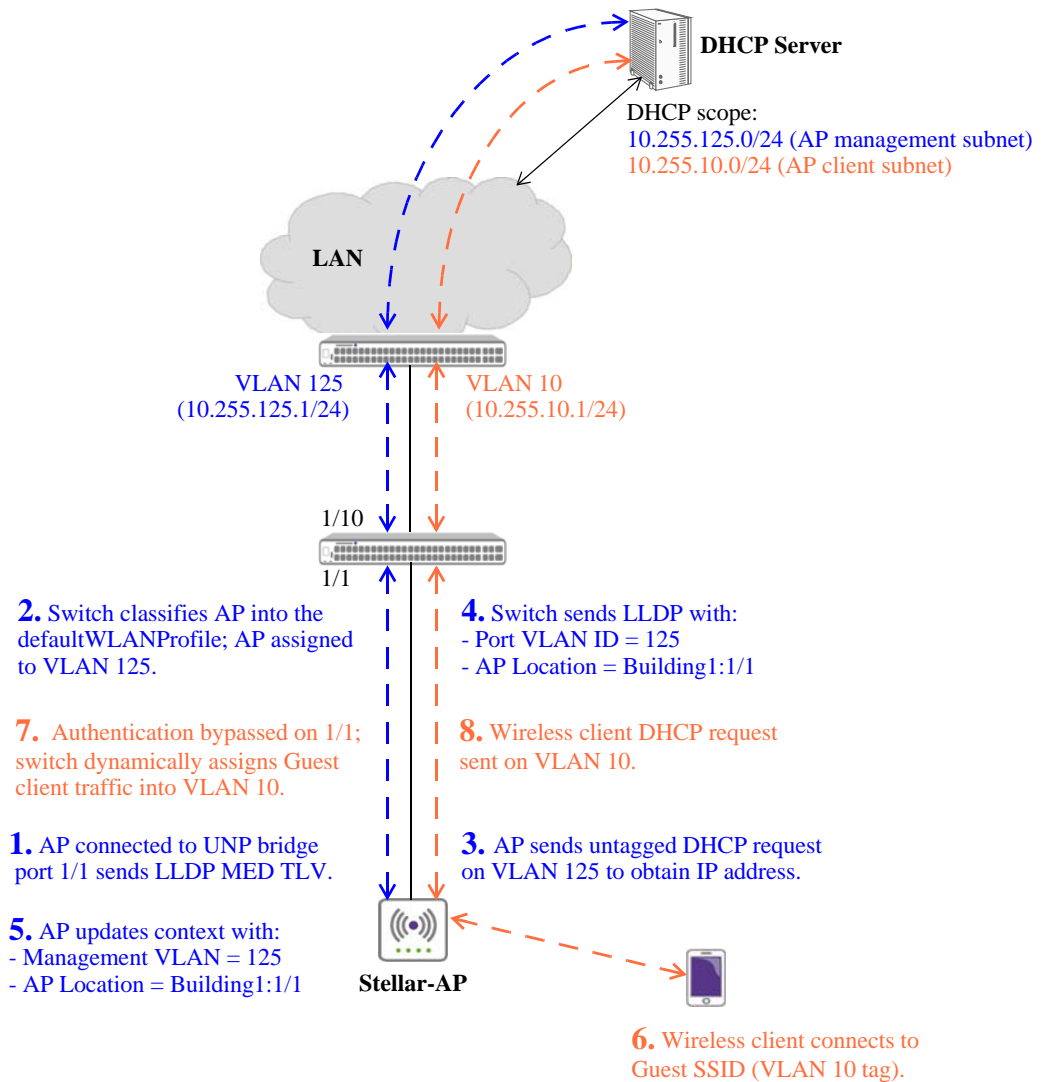
How it Works

The OmniSwitch boots up with specific default configuration and operational settings that trigger the following process to detect, learn, and classify connected Stellar AP devices:

- 1** The switch and any Stellar AP device that is connected to an 802.1x port initially exchange Link Layer Detection Protocol (LLDP) TLV packets. Through this exchange of LLDP packets, the switch identifies and learns the device MAC address as an AP.
- 2** The detection of an AP device on an 802.1x port triggers the following actions that will automatically change the operational status of the specified options (the operational status overrides the configured status).
 - The transmission of LLDP Port VLAN ID and AP Location TLVs is operationally enabled on the 802.1x port.
 - Authentication is bypassed for subsequent unknown tagged client MAC addresses that are received on the AP-detected port. The VLAN tag of the client traffic is trusted.
 - The global status for dynamic VLAN configuration is operationally enabled for the switch.
- 3** Once the AP MAC address is detected and learned, a built-in LLDP UNP classification rule for access points classifies the AP device into a built-in default profile (defaultWLANProfile). The profile is associated with a VLAN into which the AP device is classified. This establishes a VLAN-port association (VPA) between the 802.1x port and profile VLAN on which the AP MAC address is learned and forwarded.
- 4** After the AP device connection is established, classified, and the management VLAN assigned, any of the following actions can occur:
 - The AP device sends DHCP packets.
 - The switch transmits LLDP packets to the AP device to advertise the management VLAN and AP location information.
 - The AP device starts to send client-tagged traffic (tagged with the SSID VLAN). The switch will trust the VLAN tag of the AP client traffic and attempt to assign the traffic to a switch VLAN that matches the tag of the client traffic. If a matching switch VLAN does not exist, then the switch will dynamically create the necessary VLAN on which to forward the AP client traffic.
- 5** MVRP will then propagate the VLAN configuration (AP management VLAN and any static or dynamic VLAN that was automatically tagged to carry AP client traffic) to adjoining switches in the network. This process creates specific VLAN domains through which the untagged AP management traffic and tagged wireless client traffic is forwarded on the wired network.

The OmniSwitch detection and integration of OmniAccess Stellar APs results in a switch configuration that includes a management VLAN for the AP device and additional VLANs for wireless client-tagged traffic that is forwarded by the AP onto the wired network.

The following diagram shows an example of a network topology in which a Stellar AP connected to an OmniSwitch serves as a bridge between a wireless and wired network:



OmniSwitch AP Discovery and Integration Example

Configuration Guidelines

Consider the following guidelines when configuring the OmniSwitch feature components that are required to discover and integrate wireless traffic that is forwarded by OmniAccess Stellar AP devices into an OmniSwitch network:

- **Link Layer Detection Protocol (LLDP) parameters.** The first packet a connected AP device sends should be an LLDP-MED TLV that identifies the device as an AP. When the AP device is detected on the UNP port, the switch sends LLDP packets to the AP device to communicate the management VLAN (LLDP Port Vlan ID TLV) and the AP Location (LLDP Proprietary TLV).
 - The management VLAN advertised to the AP device is the VLAN associated with the UNP profile to which the AP device is classified.
 - The AP Location advertised to the AP device is derived from local switch information (such as the UNP port, chassis MAC address, system name, system location).
- **802.1x port parameters.** The port to which an AP device connects must be configured as an 802.1x port. When an AP MAC address is detected on an 802.1x port, the following occurs:
 - The switch will flush all other MAC addresses previously learned on the 802.1x port. This ensures that the AP MAC address is always the first MAC address learned on that port; a requirement that designates the UNP port as an AP-detected port.
 - The AP MAC address is authenticated based on the 802.1x port configuration.
 - Any subsequent unknown tagged client MAC addresses received are *not* authenticated. The VLAN tag of the client traffic is automatically trusted and is used to determine the OmniSwitch VLAN on which the wireless client traffic is forwarded.
- **WLAN access role profile (defaultWLANProfile).** The defaultWLANProfile is a built-in profile that is designated for classifying Stellar AP devices. This profile is automatically assigned to a built-in LLDP classification rule for APs that will recognize active AP devices connected to the switch and assign them to the defaultWLANProfile. The VLAN that is mapped to this profile will serve as the management VLAN for the classified AP devices.
 - The LLDP UNP classification rule for APs and the defaultWLANProfile are both *implicitly* configured on the switch. However, mapping a VLAN to the defaultWLANProfile requires *explicit* configuration.
 - If a Stellar AP device does not match any other Group Mobility classification rules on the switch, then the built-in LLDP rule for APs is applied and the device is assigned to defaultWLANProfile.
 - Using the defaultWLANProfile to classify AP devices ensures that all of the AP devices connected to each switch in the wired network will use the same management VLAN.
 - The defaultWLANProfile is similar to a standard UNP VLAN profile except that the profile cannot be deleted; it is a built-in profile that is always available in the switch configuration.
 - The defaultWLANProfile does not appear in the configuration snapshot for the switch. However, when the default value for any of the configurable profile attributes is modified, then the profile settings will appear in the configuration snapshot.
- **WLAN access role profile (defaultWLANProfile) attributes.** Only the following profile attributes are configurable for the defaultWLANProfile:
 - **VLAN mapping.** By default, there is no VLAN ID assigned to the built-in defaultWLANProfile. The VLAN mapping must be initially defined and can be changed at any time to a different VLAN ID if necessary.
 - **QoS policy list.** By default, there is no policy list assigned to a profile. Optionally assign a QoS policy list to apply further network access control to an AP device.
- **Dynamic VLAN configuration.** The switch operationally enables dynamic VLAN configuration to ensure that when the VLAN tag of AP client-tagged traffic does not match an existing switch VLAN, the switch will dynamically create the VLAN.
 - Client-tagged traffic forwarded from a connected AP device is classified into the VLAN that matches the VLAN tag of the client traffic. However, if that VLAN does not exist on the switch configuration, a dynamic VLAN is created. For example, if the client traffic is tagged with VLAN

- 200 but this VLAN does not exist, the switch will dynamically create VLAN 200 to accommodate the client-tagged traffic.
- Dynamic VLANs created by UNP are identified as a separate type of VLAN; the default name is set to “UNP-DYN-VLAN” and the designated type is set to “UNPD”.
 - The switch will automatically remove dynamically created VLANs when all the MAC addresses learned on the VLAN have aged out.
- **Multiple VLAN Registration Protocol (MVRP).** By default, MVRP is disabled for the switch. Enable this protocol to ensure the propagation of the AP management VLAN and any VLAN (static or dynamic) that was automatically tagged to carry AP client traffic to other switches.
 - AP management VLANs dynamically created by MVRP are converted to UNP dynamic VLANs (type UNPD).
 - If an AP client tag matches a VLAN that was dynamically created by MVRP, then that VLAN is converted to a UNP dynamic VLAN (type UNPD).

OmniAccess Stellar AP Configuration Guidelines

The Stellar AP device must meet the following configuration and operational requirements to ensure successful discovery and integration into an OmniSwitch network:

- The AP device must connect to an OmniSwitch 802.1x port. This triggers the Access Guardian process to detect and integrate the AP device.
- The first packet a connected AP device sends (before an 802.1X or DHCP packet) must be an LLDP frame with the “WLAN AP” bit set in the System Capabilities TLV and the LLDP media TLV device type set as 4(“Endpoint class 1V”). If this requirement is not met, the device may not get properly identified as an AP; this could trigger a different process for classifying the device MAC address or cause the address to be filtered.
- The AP device should always get an IP address using DHCP. Configure the DHCP server to use Option 138 for the scope of IP addresses that will be assigned to AP devices.
- AP management traffic is always sent untagged on the management VLAN advertised by the switch.
- The wireless client traffic forwarded by the AP is always tagged with the assigned SSID VLAN. Access Guardian will automatically assign the client traffic to switch VLANs that correspond to the VLAN tags of the client traffic.

Quick Steps for Configuring OmniSwitch AP Discovery

The following procedure provides a brief tutorial for configuring existing OmniSwitch features to discover and interact with OmniAccess Stellar APs.

- 1** Create the VLAN that will serve as the AP management VLAN on each participating switch in the network. For example:

```
-> vlan 125 name "AP Management VLAN"
```

- 2** Tag switch ports that connect to other switches with the VLAN created in Step 1. For example,

```
-> vlan 125 802.1q 1/10
```

- 3** Map the VLAN created in Step 1 to the built-in defaultWLANProfile. For example:

```
-> aaa user-network-profile name defaultWLANProfile vlan 125
```

4 Configure any switch port that will connect to a Stellar AP device as an 802.1x port. This requires first configuring the port as a mobile port then enabling 802.1x on the port. For example:

```
-> vlan port mobile 1/1
-> vlan port 1/1 802.1x enable
```

5 Enable MVRP for the switch to facilitate the propagation of the AP management VLAN and AP client VLANs. For example:

```
-> mvrp enable
```

6 Optionally configure a QoS policy list for the defaultWLANProfile (only the QoS policy list and VLAN mapping parameters are configurable for this profile). For example:

```
-> aaa user-network-profile name defaultWLANProfile vlan 125 policy-list-name
qlist1

-> aaa user-network-profile name defaultWLANProfile vlan 125 maximum-egress-
bandwidth 10
ERROR: Built-In Profile: defaultWLANProfile parameters cannot be modified. Vlan,
Policy-list-name are only allowed
```

7 Optionally configure the system name, system location, and port alias. The information from one or more of these settings is used to derive the AP Location information that is transmitted by the switch to the connected AP device.

```
-> system name BWIAPS01
-> system location BWI Airport Hotel
-> interfaces 1/2 alias BWI-AP01
```

Verify the OmniSwitch Configuration

1 Use the **show vlan** command to display the VLAN created for AP management and other VLANs created to carry AP client-tagged traffic. Note that “UNP-DYN-VLAN” identifies VLANs dynamically created for AP client traffic. For example:

```
-> show vlan

          stree          mble  src
vlan  type  admin  oper  1x1  flat  auth  ip  tag  lrn  name
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
   1   std   on     on    on   on    off   on  off  on  VLAN 1
  11   std   on     on    on   on    off   off off  on  VLAN 11-APclient
 200   std   on     on    on   on    off   off off  on  AP Mngmt VLAN
 500  unpd   on     on    on   on    off   off off  on  UNP-DYN-VLAN
 501  unpd   on     on    on   on    off   off off  on  UNP-DYN-VLAN
```

2 Use the **show vlan port** command to verify the port assignments for the AP management VLAN. For example:

```
-> show vlan 125 port
port      type      status
-----+-----+-----
 1/1      qtagged   inactive
 1/10     qtagged   inactive
```



```

Up Time:      11 days 3 hours 5 minutes and 49 seconds,
Contact:      Lab Admin,
Name:         BWIAPS01,
Location:     BWI Airport hotel,
Services:     78,
Date & Time:  THU JAN 22 2015 07:44:07 (UTC)
Flash Space:

```

```

Primary CMM:
  Available (bytes): 1121243136,
  Comments           : None

```

```

-> show interfaces 1/1 port
Legends: WTR - Wait To Restore
# - WTR Timer is Running & Port is in wait-to-restore state
* - Permanent Shutdown

```

| Slot/ Port | Admin Status | Link Status | Violations | Recovery Time | Recovery Max | WTR (sec) | Alias |
|---------------|-----------------|----------------|------------|------------------|-----------------|--------------|------------|
| 1/10 | enable | down | none | 300 | 10 | 0 | "BWI-AP01" |

7 Use the **show 802.1x ap-client-mac** command to display all the wireless client MAC addresses received on an 802.1X port that is connected to an OmniAccess Stellar AP. For example:

```

-> show 802.1x AP-client-mac
Slot  MAC              Vlan
Port  Address             Learned
-----+-----+-----
1/1   00:00:c3:de:79:b8  200
      00:00:c3:de:80:b1  210
1/20  00:00:c3:d3:81:c1  300
      00:00:c3:d3:82:c2  310

```

Verifying Access Guardian Users

The following set of **show aaa-device** commands provide a centralized way to verify the status of users authenticated and classified through Access Guardian security mechanisms:

1 The **show aaa-device all-users** command displays the Access Guardian status of all users learned on 802.1x ports:

```
-> show aaa-device all-users
```

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|--------------|---------------------|--------|-----------------|--------------|
| 1/1 | 00:11:50:a6:12:00 | User101 | 100 | Brdg | 10.133.0.100 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:01 | User101 | 100 | Brdg | 10.133.0.101 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:02 | User101 | 100 | Brdg | 10.133.0.102 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:03 | User101 | 100 | Brdg | 10.133.0.103 | 1X | Pass | Marketing | |
| 1/1 | 00:1a:50:a6:12:50 | -- | 100 | Blk | 10.133.2.128 | None | N/A | enr_no_internet | |
| 1/1 | 00:1a:50:a6:12:51 | -- | 100 | Blk | 10.133.2.129 | None | N/A | enr_no_internet | |
| 1/1 | 00:1a:50:a6:12:52 | -- | 100 | Blk | 10.133.2.130 | None | N/A | enr_no_internet | |
| 1/1 | 00:1a:50:a6:12:53 | -- | 100 | Blk | 10.133.2.131 | None | N/A | enr_no_internet | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 1/2 | 00:00:39:47:4f:0c | pc2006 | 1000 | Brdg | - | 1X | Pass | Marketing | |
| 1/2 | 00:b0:d0:77:fa:72 | -- | 1000 | Brdg | - | MAC | Pass | Marketing | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 5/9 | 00:90:27:17:91:a8 | pc2006 | 1000 | Brdg | - | 1X | Pass | enr | |
| 5/9 | 00:00:39:93:46:0c | -- | 1 | Blk | - | MAC | Fail | - | |

2 The **show aaa-device supplicant-users** command displays the Access Guardian status of all supplicant (802.1x) users learned on the switch:

```
-> show aaa-device supplicant-users
```

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|--------------|---------------------|--------|--------------|--------------|
| 1/1 | 00:11:50:a6:12:00 | User101 | 100 | Brdg | 10.133.0.100 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:01 | User101 | 100 | Brdg | 10.133.0.101 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:02 | User101 | 100 | Brdg | 10.133.0.102 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:03 | User101 | 100 | Brdg | 10.133.0.103 | 1X | Pass | Marketing | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 1/2 | 00:00:39:47:4f:0c | pc2006 | 1000 | Brdg | - | 1X | Pass | Marketing | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 5/9 | 00:90:27:17:91:a8 | pc2006 | 1000 | Brdg | - | 1X | Pass | enr | |
| 5/9 | 00:00:39:93:46:10 | -- | 1 | Blk | - | 1X | Fail | - | |

3 The `show aaa-device non-supPLICANT-users` command displays the Access Guardian status of all non-supPLICANT (non-802.1x) users learned on the switch:

```
-> show aaa-device non-supPLICANT-users
```

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|--------------|---------------------|--------|-----------------|--------------|
| 1/1 | 00:1a:50:a6:12:50 | -- | 100 | Blk | 10.133.2.128 | None | N/A | enr_no_internet | |
| 1/1 | 00:1a:50:a6:12:51 | -- | 100 | Blk | 10.133.2.129 | None | N/A | enr_no_internet | |
| 1/1 | 00:1a:50:a6:12:52 | -- | 100 | Blk | 10.133.2.130 | None | N/A | enr_no_internet | |
| 1/1 | 00:1a:50:a6:12:53 | -- | 100 | Blk | 10.133.2.131 | None | N/A | enr_no_internet | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 1/2 | 00:b0:d0:77:fa:72 | -- | 1000 | Brdg | - | MAC | Pass | Marketing | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 5/9 | 00:90:27:17:91:20 | pc2006 | 1000 | Brdg | - | MAC | Pass | enr | |
| 5/9 | 00:00:39:93:46:0c | -- | 1 | Blk | - | MAC | Fail | - | |

4 The `show aaa-device captive-portal-users` command displays the Access Guardian status of all users that attempted network access through the switch using Captive Portal web-based authentication:

```
-> show aaa-device captive-portal-users
```

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|--------------|---------------------|--------|--------------|--------------|
| 1/1 | 00:11:50:a6:12:00 | User101 | 100 | Brdg | 10.133.0.100 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:01 | User101 | 100 | Brdg | 10.133.0.101 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:02 | User101 | 100 | Brdg | 10.133.0.102 | 1X | Pass | Marketing | |
| 1/1 | 00:11:50:a6:12:03 | User101 | 100 | Brdg | 10.133.0.103 | 1X | Pass | Marketing | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 1/2 | 00:00:39:47:4f:0c | pc2006 | 1000 | Brdg | - | 1X | Pass | Marketing | |
| 1/2 | 00:b0:d0:77:fa:72 | -- | 1000 | Brdg | - | MAC | Pass | Marketing | |

| Slot Port | MAC Address | User Name | VLAN | Addr Mode | IP Address | Authentication Type | Result | User Profile | Network Name |
|-----------|-------------------|-----------|------|-----------|------------|---------------------|--------|--------------|--------------|
| 5/9 | 00:90:27:17:91:a8 | pc2006 | 1000 | Brdg | - | 1X | Pass | enr | |
| 5/9 | 00:00:39:93:46:0c | -- | 1 | Blk | - | MAC | Fail | - | |

For more information about the displays that result from these commands, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Logging Users out of the Network

In the event that it becomes necessary to manually log a user out of the network, the **aaa admin-logout** command is available to the switch admin user. The following parameters are available with this command to specify which users to log out:

- **mac-address**—Logs out the user device with the specified source MAC address. For example:

```
-> aaa admin-logout mac-address 00:2a:95:00:3a:10
```
- **port slot/port**—Logs out all users connected to the specified slot and port number. For example:

```
-> aaa admin-logout port 1/9
```
- **user user_name**—Logs out the user device accessing the network with the specified user name account. For example:

```
-> aaa admin-logout user j_smith
```
- **user-network-profile name profile_name**—Logs out all users classified with the specified profile name. For example:

```
-> aaa admin-logout user-network-profile name marketing
```

Logging a group of users out of the network is particularly useful if configuration changes are required to any Access Guardian features. For example, if the Host Integrity Check (HIC) feature is globally disabled for the switch, all User Network Profiles (UNP) with the HIC attribute enabled no longer check devices for compliance. This could allow users that don't comply with security requirements to access the network. The solution:

- 1 Log out all users associated with the profile using the **aaa admin-logout** command.
- 2 Disable the HIC feature for the switch using the **aaa hic disable** command.
- 3 Make any necessary configuration changes to the HIC feature (for example, add a remediation server to the HIC exception list).
- 4 Enable the HIC feature for the switch using the **aaa hic enable** command. When HIC is enabled, all users associated with the HIC-enabled UNP are checked for compliance.

Note. The **aaa admin-logout** command is only available to the switch admin user. The admin account, however, is protected from any attempts to log out the admin user.

For more information about HIC and user profiles, see “[Host Integrity Check \(End-User Compliance\)](#)” on page 35-19 and “[User Network Profiles \(Role-Based Access\)](#)” on page 35-22.

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 classification-rule | Displays the UNP mobile classification rule configuration for the switch. |
| show aaa classification-rule | Displays the global Host Integrity Check (HIC) configuration for the switch. |
| show aaa hic host | Displays a list of the learned host MAC addresses and the HIC status for each host. |
| show aaa classification-rule | Displays the HIC server configuration for the switch. |
| show aaa hic server-failure policy | Displays the Host Integrity Check (HIC) server exception list. |
| show aaa classification-rule | 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 AOS Release 6 CLI Reference Guide*.

Bring Your Own Device (BYOD) Overview

The OmniSwitch implementation of Bring Your Own Devices (BYOD) leverages the OmniVista Unified Policy Access Manager (UPAM) or ClearPass Policy Manager (CPPM) and Access Guardian features on the OmniSwitch. BYOD can be implemented on a campus, branch offices, Internet edge, and converged access networks. It allows a wired or wireless guest, device, or authenticated user to connect to the network through an OmniSwitch edge device using the UPAM or CPPM for unified authentication.

The BYOD support on OmniSwitch provides the following:

- Unified access policy management solution for Wireline and Wireless networks using UPAM or CPPM
- Integration with Access Guardian UNPs and 802.1x authentication.

Note. For additional information, refer to the following:

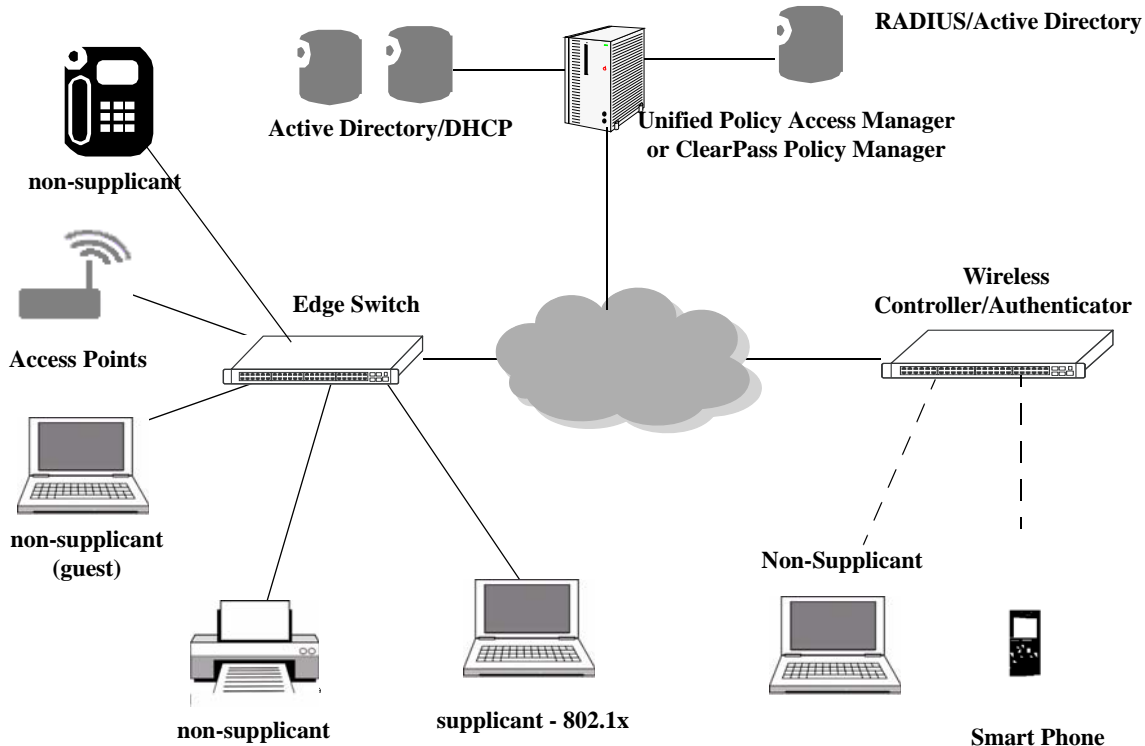
- [“Configuring User Network Profiles” on page 35-51](#) and [Chapter 41, “Configuring 802.1X.”](#) for additional information about UNPs and 802.1x authentication
- OmniAccess WLAN documentation.
- OmniVista Unified Policy Access Manager documentation for in-depth OmniSwitch and server configuration requirements.
- ClearPass Policy Manager documentation for in-depth server configuration and licensing requirements.

-
- RADIUS Change of Authorization (CoA):
 - Provides a mechanism to change AAA attributes of a session after authentication.
 - Sends the New Profile as an attribute in the message.
 - Sends a Disconnect Message to terminate user session and discard all user context.
 - A validated BYOD solution using UPAM or CPPM with CoA and the OmniSwitch.
 - Restricted access to the network and validation for end user devices including employees with IT supplied devices, IP phones, employees personal devices, guest devices, access points, cameras, and silent devices such as printers.
 - UPAM or CPPM can act as a RADIUS server for new deployments and/or a RADIUS proxy for existing networks.
 - Captive portal redirect using a new dynamic redirect URL Vendor Specific Attribute (VSA).

Key Components of a BYOD Solution

The OmniSwitch BYOD solution comprises of the following main components:

- The network infrastructure consisting of both wireless and wireline network. OmniSwitch leverages the Access Guardian features such as 802.1x supplicants, non-suppliant MAC authentication, and user network profiles (UNP) to support the BYOD solution.
- The UPAM or CPPM interacts with both wireless and wireline networks acting as a RADIUS server and/or RADIUS server proxy. The UPAM or CPPM provides policy management, guest access, on-boarding, and posture checking capabilities.



BYOD Network Illustration

ClearPass Policy Manager

ClearPass Policy Manager (CPPM) association and configuration is required for the OmniSwitch BYOD solution. This section describes the various services, features, and settings provided by ClearPass and interaction with OmniSwitch.

Note. The information provided in this section is specific to CPPM. For information about the Unified Policy Access Manager (UPAM), refer to the OmniVista UPAM documentation.

ClearPass Guest

The OmniSwitch BYOD solution supports guest self registration, sponsored guest access and pre-registration of guest devices using MAC and Captive Portal authentication.

- Self Registration
 - An integrated external Captive Portal for guest or visitor registration.
 - Redirection to a customizable guest registration Captive Portal
- Sponsored Access
 - SMS and text email notification

ClearPass Policy Manager

ClearPass provides a user and device-independent framework that supports any BYOD initiative, large or small, by providing:

- Self-service on-boarding, provisioning and revocation of access for all major mobile devices.
- Device profiling as a basis for grooming traffic and improving network security based on device category such as:
 - Device Category - Computer, Printer, AP
 - OS Family - MAC, Android, Windows, Linux
 - Device name and OS version
 - Useful for wired devices such as printers, access points, IP Phones, and cameras
- Controlled access and remediation for compromised devices
- Device disconnect if device signature changes
 - Secure guest network access with simplified workflows.

ClearPass Onboard

The BYOD solution supports the following services for device on-boarding and device management for guest and registered devices:

- Automatic configuration of Wireless, Wired 802.1X, VPN settings of personal and corporate devices that are connecting to the network for the first time.
- Management of digital certificates.
- Device on-boarding system is integrated with the External Captive portal that is separate from OmniSwitch captive portal.

- Integration with Enterprise Active Directory for authentication of employee credentials before issuing device credentials.
- Device provisioning supported through Aruba Quick Connect or Apple OTA API.
- Quick Connect supports native supplicants on Windows Vista, XP, 7, Apple, and Android devices.

ClearPass OnGuard

ClearPass OnGuard agents perform advanced endpoint posture checking to ensure compliance is met before the devices connect. OnGuard has the following functionalities:

- Enhanced capabilities for endpoint compliance and control.
- Supports Microsoft, Apple, and Linux operating systems.
- Anti-virus, anti-spyware, firewall checks and more using the persistent or dissolvable agent.
- Optional auto-remediation and quarantine capabilities.
- System-wide endpoint messaging, notifications and session control.
- Centrally view the Online status of all devices from the ClearPass Policy Manager platform.

OmniSwitch Integration with UPAM or CPPM for BYOD Support

Consider the following key points regarding OmniSwitch integration with UPAM or ClearPass for BYOD support:

- The same UNPs and access lists must be configured on both OmniSwitch and UPAM or CPPM for proper alignment.
- RADIUS server configured on OmniSwitch must point to UPAM or CPPM in both proxy and server cases.
- A redirection server must be configured on OmniSwitch that points to UPAM or CPPM.
- Dynamic Vendor Specific Attribute (VSA) URL redirect can be implemented using the OmniSwitch VSAs. The VSAs must be downloaded and installed on the ClearPass server; refer to the OmniVista UPAM documentation for information about how VSAs are installed on the UPAM server.
- Port bounce capability can be configured on the OmniSwitch to ensure a clean re-authentication process for non-supPLICANT devices.
- A PAUSE timer can be configured that flushes out a user context (that is used for a welcome page or other user context information) on timer expiry.

RFC-3576 Attributes

RADIUS servers and the OmniSwitch can be configured with particular attributes defined in RFC 3576. These attributes carry specific authentication, authorization, and configuration details about RADIUS requests to and replies from the server. This section describes the attributes specific to an OmniSwitch BYOD solution.

| Num. | CoA Attribute | Notes |
|-------------|----------------------|--------------|
|-------------|----------------------|--------------|

| | |
|------------------------------|--|
| 40 Disconnect-Request | <p>Disconnect Request sent by RADIUS/UPAM or ClearPass server.</p> <ul style="list-style-type: none"> • The Disconnect-Request RADIUS message contains the User-Name or the Calling-Station-ID attribute. • When the message contains both the User-Name and Calling-Station-ID, the MAC address is identified based on the Calling-Station-ID only. |
| 41 DM-ACK | <p>On reception of Disconnect request message (DM), all device authentication is removed from the switch. Disconnect request message (DM) Acknowledgment for RADIUS/UPAM or ClearPass authentication.</p> |
| 42 DM-NACK | <p>Disconnect request message (DM) Not Acknowledged</p> |
| 43 CoA-Request | <p>CoA message is sent from UPAM or ClearPass Server. CoA-Request packets contain information for dynamically changing session authorizations. The following attributes are used:</p> <ul style="list-style-type: none"> • The User-Name: AOS retrieves the MAC address associated to this user • The Calling-Station-ID: This explicitly specify the user MAC address <p>When the message contains both the User-Name and Calling-Station-ID, the MAC address is identified based on the Calling-Station-ID only.</p> |
| 44 CoA-ACK | <p>Supports a Change of Authorization-Request (CoA) message for RADIUS authentication. COA-ACK is sent by OmniSwitch to UPAM or ClearPass that has attributes MD5 hash value and Identifier.</p> |
| 45 CoA-NACK | <p>COA-NACK message is sent from OmniSwitch. For NAK message, the Error-Cause attribute must be supported and filled accordingly.</p> |
| Error-Cause | <p>Supported as part of CoA-NAK and DM-NAK message. Error-Cause Scenarios: Missing Attribute - If User name and Calling station ID Filter ID not present Invalid Request - If Client context does not exist</p> |

Vendor-Specific Attributes for UPAM or ClearPass

The OmniSwitch 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).

For UPAM or ClearPass integration, the VSA dictionary must be updated with the "**Alcatel-Redirect-URL**" and the "**Alcatel-Access-Policy-List**" VSA that can be imported into the UPAM or ClearPass server. The following VSAs can be imported to the UPAM or ClearPass server:

| Num. | ClearPass/RADIUS VSA | Type | Description |
|------|----------------------------|--------|---|
| 6 | Alcatel-Port-Desc | string | <p>Description of the port. This attribute is currently defined in the Alcatel dictionary as:</p> <p>RADIUS attribute type = 26 (VSA) VSA Vendor ID = 800 VSA Type = 26 VSA format = string</p> <p>This attribute is included in all RADIUS messages sent by Alcatel-Lucent OmniSwitch (Access-Request, Accounting-Request Start, Accounting-Request Interim and Accounting-Request Stop).The attribute is set with the alias configured for the port. When the alias is not set, VSA will be an empty string.</p> |
| 100 | Alcatel-Access-Policy-List | string | Configures UPAM or ClearPass to the policy list associated with the UNP. |
| 101 | Alcatel-Redirect-URL | string | Configures UPAM or ClearPass to send redirection URL as part of RADIUS response redirecting the user web traffic. |
| 101 | Redirection-Status | string | Specifies Redirect Status |

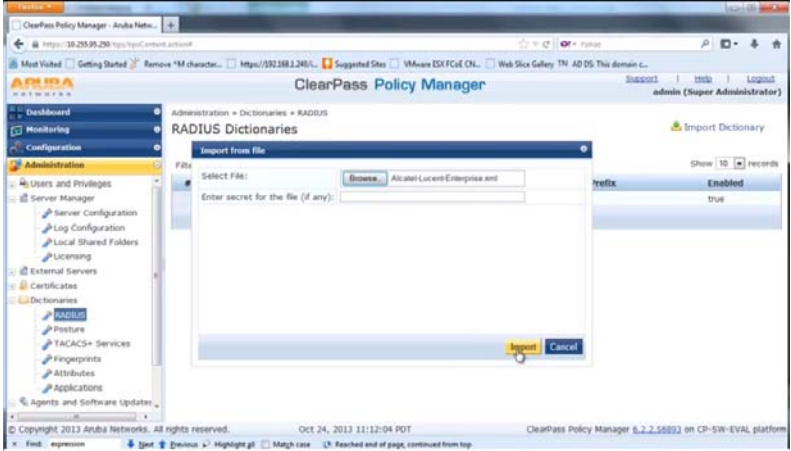
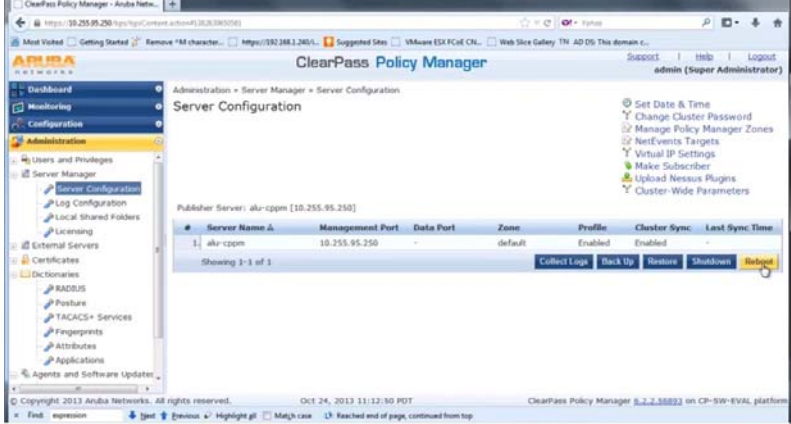
Importing the Alcatel-Lucent dictionary into CPPM

Note. The procedure described in this section is specific to importing the VSA dictionary into the CPPM server. For information about the VSA dictionary integration with UPAM, refer to the OmniVista UPAM documentation.

Perform the following to import the VSA dictionary into the CPPM server:

- 1 Download the **Alcatel-Lucent-Enterprise.xml** file from the Service & Support website.
- 2 Click on **Dictionary->Import Dictionary** and browse for the **Alcatel-Lucent-Enterprise.xml** file.
- 3 Click on **Server Configuration->Reboot** to reboot the server.

Importing the Alcatel-Lucent dictionary into CPPM

| | |
|---|--|
| <p>Import Dictionary: Dictionary->Import Dictionary</p> |  |
| <p>Reboot CPPM Server Configuration->Reboot</p> |  |

Port Bounce

A port bounce is used to terminate a user session and discard all associated session context for non-suplicants. This is done by disabling and re-enabling the port and clearing any authentication state for the devices on the port. A port bounce action is configurable through the **aaa port-bounce** command.

- Port bounce is used for MAC authenticated non-suppliant users.
- On receipt of Disconnect Request or CoA message, the OmniSwitch determines if the user needs to move or change VLANs. The switch clears the device authentication state information and waits a configurable amount of time prior to allowing the device to re-authenticate.
- If the new UNP specifies a different VLAN ID, the port bouncing feature is enforced as per configuration for non-suplicants.
- When a device changes VLANs and it is the only device on the port, the switch port is bounced to ensure a clean reconnection and get the correct IP address through DHCP.
- Port bouncing is enforced only if the non-suppliant user is the only user on the port. Also if a CoA message is received for a non-suppliant user and port bouncing is disabled globally but is enabled on the port on which the non-suppliant user has been classified, then the port is bounced.

Note. During CoA packet processing, if the UNP returned in both level of authentication is from the same VLAN then the Policy-List returned from the server is processed.

If the server returned Policy-List is not available in the switch then the Policy-List associated with the UNP is applied.

Pause timer

Based on the port bouncing logic, the switch clears all authentication states of the device by pausing for some period of time. The value for the period of time is configurable through the **aaa redirect pause-timer** command.

When supplicant devices are detected, the switch must clear all authentication states on the device and pause for some period of time before redirection to the specified URLs. The pause mechanism is enforced when the following conditions are met.

- COA message received by the switch indicates VLAN movement for the non-suppliant user, and
- Port bouncing is disabled for the user port or UNP.

The pause mechanism ensures that all traffic from the user is dropped until the global pause timer expires and the corresponding user context is removed. This process triggers re-authentication of the user.

Note. The Port Bounce and Pause Timer functions apply only to non-suppliant devices. For supplicant devices, there is no difference whether Port Bounce is enabled or Pause Timer is enabled. The user context for supplicant devices is removed by triggering the re-authentication of the supplicant user and the device moves into a new UNP profile and VLAN after re-authentication.

Configuring OmniSwitch BYOD Support

BYOD is supported on 802.1x ports for supplicant and non-supplicant registered and guest users and devices. The BYOD solution leverages the existing Access Guardian UNP capability and is applicable only on 802.1x ports. The following generic configuration examples apply only to the OmniSwitch components for interaction with a UPAM or CPPM server. For more detailed application examples, refer to [“BYOD Application Examples” on page 35-86](#).

Note. Configure the OmniSwitch to interact only with the OmniVista UPAM server or the CPPM server.

Configuring the UPAM or CPPM server as an AAA RADIUS Server

The server and the authentication types must be configured to allow the OmniSwitch to forward authentication requests to the UPAM or CPPM. For example:

```
-> aaa radius-server "cppm" host 1.1.1.1 key e47ac0f11e9fa869 retransmit 3
timeout 2 auth-port 1812 acct-port 1813
-> aaa authentication 802.1x cppm
-> aaa authentication mac cppm
-> aaa accounting 802.1x cppm
-> aaa accounting mac cppm
```

Configuring 802.1x

802.1x supplicant and non-supplicant settings must be enabled on the ports for the authentication process to begin. Configure the port as a mobile and 802.1X port using the [vlan](#) and [802.1x](#) commands. For example:

```
-> vlan port mobile 1/4
-> vlan port 1/4 802.1x enable
-> 802.1x 1/4 supplicant policy authentication pass block fail block
-> 802.1x 1/4 non-supplicant policy authentication pass block fail block
```

Configuring Redirection with Dynamic URLs

The redirect server and the URL returned by the server are used to present guest users with different web pages depending on what state of authentication they are in. Use the [aaa redirect-server](#) command to specify the IP address of the redirect server. For example:

```
-> aaa redirect-server CPPM ip-address 192.168.1.244 url-list url1 url2 url3
```

Configuring UNP Profiles

Profiles are used to move users into an appropriate UNP based on the authentication process. Use the [aaa user-network-profile](#) command for configuring UNPs. For example:

```
-> aaa user-network-profile name "UNP-guest" vlan 1002
-> aaa user-network-profile name "UNP-restricted" vlan 1002
```

To support interaction with the UPAM or CPPM server, the same UNP profile name must be configured on both the OmniSwitch and the UPAM or CPPM server.

Configuring Port Bounce

Port bouncing is used to force a re-authentication for non-supPLICANT devices. Use the [aaa port-bounce](#) command. For example:

```
-> aaa port bounce enable
-> aaa port bounce 1/1-5 enable
```

Configuring the Pause Timer

Use the [aaa redirect pause-timer](#) command. For example:

```
-> aaa redirect pause timer 120
```

BYOD Authentication Process Overview

This section describes the basic BYOD process with respect to the OmniSwitch and its interaction with the UPAM or ClearPass server.

Authentication for Registered Devices (802.1x)

The BYOD solution provides the following authentication process for registered devices (for example, IT issued employee devices):

- 1 When an 802.1x enabled port on OmniSwitch detects the user the authentication process is triggered to classify the user.
- 2 The OmniSwitch sends a request to the UPAM or ClearPass server that authenticates the user based on user credentials and the profiles and policies configured on the UPAM or ClearPass server.
- 3 UPAM or ClearPass classifies the user to a registered UNP and returns the UNP information to the OmniSwitch.
- 4 The OmniSwitch assigns the user to the UNP obtained from the UPAM or ClearPass server.

Authentication for Network Devices (MAC Authentication)

The BYOD solution provides the following MAC authentication process for network devices such as IP phones, printers, or access points.

- 1 When MAC authentication is enabled on a port and the OmniSwitch detects the device, MAC authentication process is triggered to classify the device.
- 2 The OmniSwitch sends a request to the UPAM or ClearPass server that authenticates the device based on the device MAC address and the profiles and policies configured on the UPAM or ClearPass server.
- 3 UPAM or ClearPass classifies the device to a UNP and returns the UNP information to the OmniSwitch.
- 4 The OmniSwitch assigns the device to the UNP obtained from the UPAM or ClearPass server.

Authentication for Guest Devices and Employee On-boarding

The BYOD solution provides the following authentication process for guest devices and employee personal devices:

- 1** When MAC authentication is enabled on a port and the OmniSwitch detects the device, the MAC authentication process is triggered to classify the device.
- 2** UPAM or ClearPass initially classifies the device to a temporary UNP and returns a redirection URL that allows for guest registration or employee on-boarding.
- 3** The OmniSwitch assigns the user to the specified UNP. Since redirection is also set, all DHCP or DNS traffic is allowed but HTTP traffic from the user is redirected towards the URL returned in the UNP.
- 4** The user is presented with a guest login page or an on-boarding page to enter user credentials.
- 5** UPAM or ClearPass determines the appropriate role of the user after performing registration and sends the final UNP to the OmniSwitch through a CoA request or RADIUS packet for on-boarding.

Zero Configuration Networking (mDNS and SSDP)

mDNS is a resolution service used to discover services on a LAN. mDNS allows resolving host names to IP addresses within small networks without the need of a conventional DNS server. The mDNS protocol uses IP multicast User Datagram Protocol (UDP) packets and is implemented by Apple Bonjour, Avahi (LGPL), and Linux NSS-mDNS. mDNS can be leveraged in a BYOD network by allowing wireless guests and visitors access to network devices such as printers.

To resolve a host name, the mDNS client broadcasts a query message asking the host having that name to identify itself. The target machine then multicasts a message that includes its IP address. All machines in that subnet will use that information to update their mDNS caches.

For example the Apple's Bonjour architecture implements the following three fundamental operations to support zero configuration networking service:

- Publication (Advertising a service)
- Discovery (Browsing for available services)
- Resolution (Translating service instance names to address and port numbers for use)

Zero configuration networking is a set of protocols that can be used to assign IP addresses, resolve names and discover services. It allows communications between network devices and allowing them to advertise and share each others' resources.

Apple device use mDNS (multicast DNS) as the underlying zeroconfiguration protocol for Bonjour exchanges. Apple devices use Bonjour to discover Services over a network. These services include AppleTV or Print services. Apple's Bonjour protocol, built on multicast DNS, is a Layer 2 non-routable protocol. This means that only clients on the same subnet as the AirPrint and AirPlay enabled devices can see those services. On a network that has multiple subnets, the multicast DNS advertisements will not reach users on different subnets. Enterprises, schools, universities and many other environments are typically built with multiple subnets, which mean that although Apple services may be available to users, they will not be visible to them.

Similarly DLNA (Digital Living Network Alliance) is a standard that is derived from UPnP (Universal Plug and Play). DLNA uses SSDP (Simple Service Discovery Protocol) for service discovery on the network. It provides the ability to share digital media services for Android or Windows devices. The SSDP protocol also has the same limitation of local subnet scope.

Hence, the zero configuration mDNS and SSDP solution is developed to extend mDNS and SSDP across Layer 3.

The zero configuration mDNS and SSDP solution allows:

- mDNS and SSDP compatible devices to discover network services across IP subnet boundaries.
- To provide the solution that is unified across wire or wireless network.

The mDNS or SSDP packet handling across layer 3 supports three mode of operation:

- **Tunnel (Aruba) Mode:** Supports mDNS or SSDP compatible devices with Aruba controller with GRE tunnel protocol type 0x0. This is the default mode of operation.
- **Tunnel Standard Mode:** Supports mDNS or SSDP compatible devices with responder with GRE tunnel protocol type 0x6558.
- **Gateway Mode:** Supports mDNS or SSDP compatible devices to discover network services across IP subnet boundaries or VLANs.

Quick Steps for Zero Configuration

The zero configuration varies with respect to the type of modes.

If the network consists of Aruba wireless controllers (Tunnel (Aruba) mode), the following must be configured on the edge switch:

1 Enable mDNS and SSDP using the **zeroconf mdns admin-state** and **zeroconf sstp admin-state** command.

To enable mDNS relay on the switch, enter:

```
-> zeroconf mdns admin-state enable
```

To enable SSDP relay on the switch, enter:

```
-> zeroconf sstp admin-state enable
```

2 Configure the mode of operation for the switch. For Aruba APs the mode must be set to tunnel. To configure the mode, use the **zeroconf mode** command. For example:

```
-> zeroconf mode tunnel
```

Note. By default the switch will be in Tunnel (Aruba) mode.

3 Configure the tunnel source IP address (Loopback0 IP interface) for the GRE tunnel. To configure the Loopback 0 IP interface, use the **ip interface** command. For example:

```
-> ip interface Loopback0 address 10.1.2.3
```

Note. If the Loopback0 address is not configured, the operational status of mDNS or SSDP will be down.

4 Configure the Aruba responder IP address. The responder IP address must be configured to tunnel the mDNS or SSDP packets in tunnel mode.

To configure the responder IP address, use the **zeroconf responder-ip** command. For example:

```
-> zeroconf responder-ip 10.0.1.5
```

Note. If the responder IP address is not configured or not reachable, then the operational status of mDNS or SSDP will be down.

If the network does not consist of any responder and uses gateway, the following must be configured on the gateway switch:

1 Enable mDNS and SSDP using the **zeroconf mdns admin-state** and **zeroconf sstp admin-state** command.

To enable mDNS relay on the switch, enter:

```
-> zeroconf mdns admin-state enable
```


To enable SSDP relay on the switch, enter:

```
-> zeroconf sstp admin-state enable
```

2 Configure the mode of operation for the switch. For gateway mode, set the mode to gateway.

To configure the mode, use the **zeroconf mode** command. For example:

```
-> zeroconf mode gateway
```

3 Configure the gateway VLAN list. Traffic from the edge switches will be forwarded at L2 to the gateway switch. From the gateway switch the mDNS and SSDP packets will be relayed to other VLANs based on the gateway VLAN list configured.

To configure the gateway VLAN list, use the **zeroconf gateway-vlan-list** command. For example:

```
-> zeroconf gateway-vlan-list 1 4 6
```

Note. Maximum of 10 gateway VLANs is supported in a list.

If the network consists of responder (Tunnel Standard mode), the following must be configured on the edge switch:

1 Enable mDNS and SSDP using the **zeroconf mdns admin-state** and **zeroconf sstp admin-state** command.

To enable mDNS relay on the switch, enter:

```
-> zeroconf mdns admin-state enable
```

To enable SSDP relay on the switch, enter:

```
-> zeroconf sstp admin-state enable
```

2 Configure the mode of operation for the switch. Set the mode to tunnel type standard.

To configure the mode, use the **zeroconf mode** command. For example:

```
-> zeroconf mode tunnel type standard
```

3 Configure the tunnel source IP address (Loopback0 IP interface) for the GRE tunnel. To configure the Loopback 0 IP interface, use the **ip interface** command. For example:

```
-> ip interface Loopback0 address 10.1.1.2.3
```

Note. If the Loopback0 address is not configured, the operational status of mDNS or SSDP will be down.

4 Configure the remote tunnel endpoint IP address of the centralized switch running the responder.

To configure the responder IP address, use the **zeroconf responder-ip** command. For example:

```
-> zeroconf responder-ip 10.0.1.5
```

5 Configure the access VLAN list. If the mDNS packet comes in with a VLAN tag of 4095, then the packet is flooded to the access VLAN list configured.

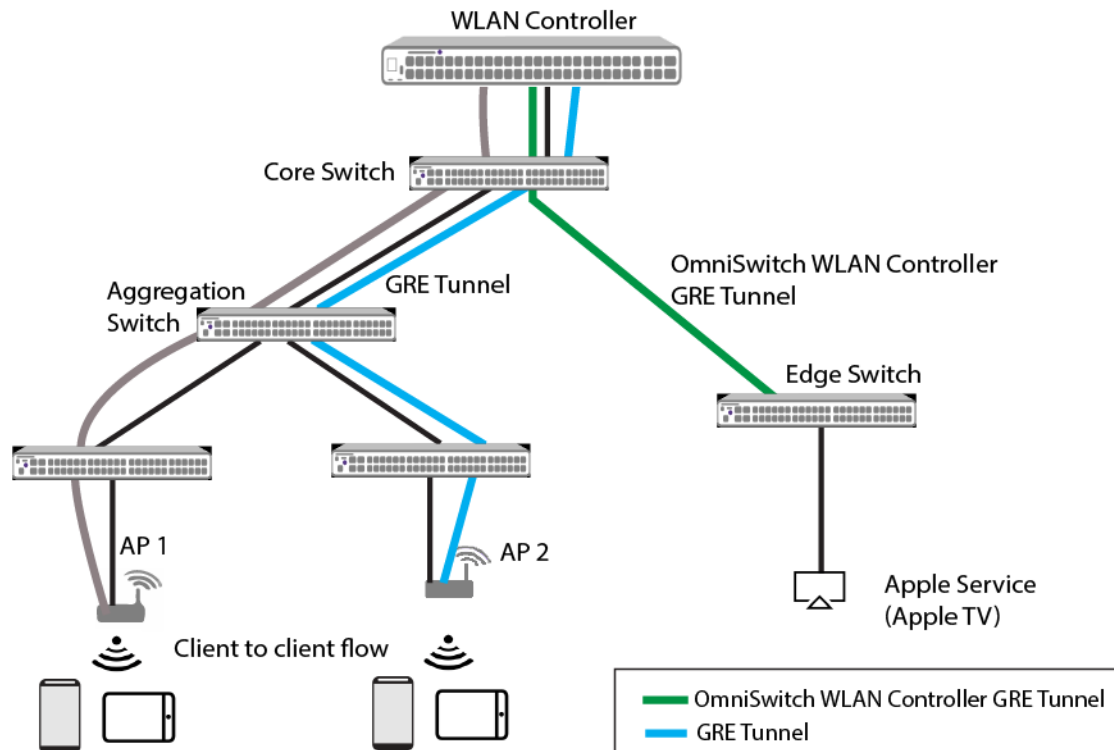
To configure the access VLAN list, use the **zeroconf access-vlan-list** command. For example:

```
-> zeroconf access-vlan-list 7 8 9
```

Note. Maximum of 16 access VLANs is supported in a list.

mDNS Work Flow

The following diagram represents a mDNS work flow setup. The wireless clients connected to Access point 1 (AP1) or Access Point 2 (AP2) request for the mDNS service offered.



mDNS Work Flow

The mDNS feature is enabled on the OmniSwitch to support the mDNS service. A Layer 2 GRE tunnel interface is configured from the WLAN controller to the OmniSwitch to relay the mDNS messages.

The mDNS message from the Bonjour capable wired service device is encapsulated and relayed from the OmniSwitch to the configured WLAN controller over the GRE tunnel. The WLAN controller then relays the mDNS messages received via the OmniSwitch GRE tunnel to the APs over the AP GRE tunnels.

Note. The WLAN controller uses a multicast optimization algorithm and forwards Bonjour response messages to targeted user devices, instead of all devices on all APs. This limits the unnecessary flooding of the Bonjour or mDNS traffic to improve the Wi-Fi performance.

Operating Principle

Tunnel (Aruba) Mode

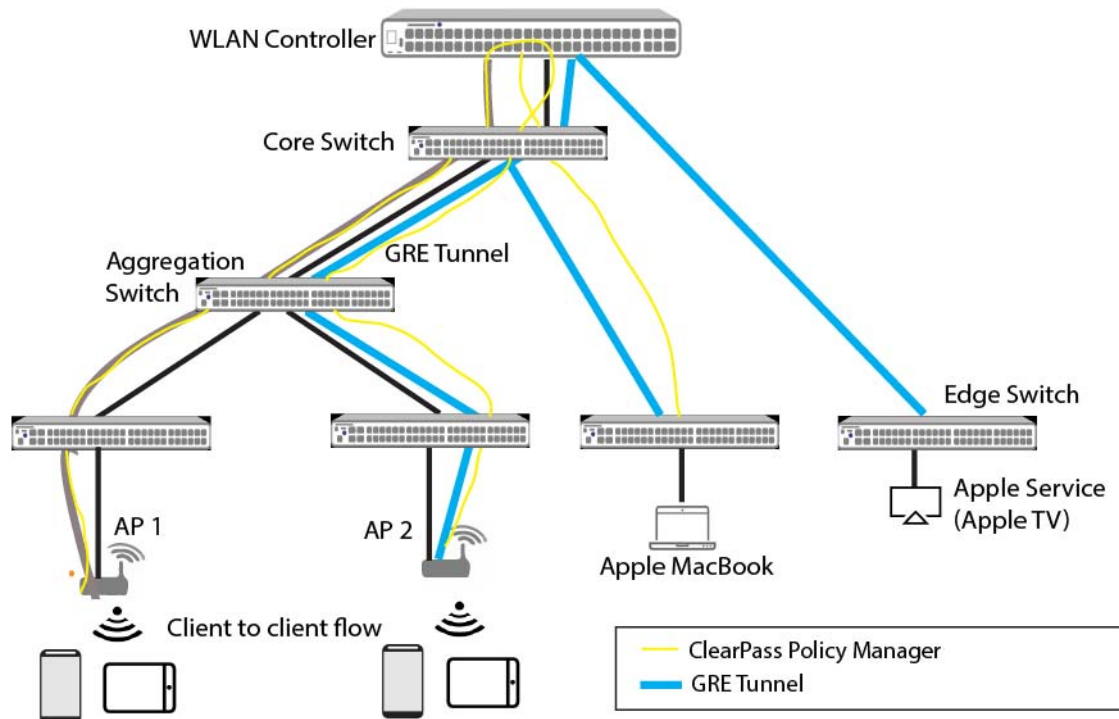
Tunnel (Aruba) Mode is configured if the network has only Aruba wireless controller. In this mode, all the switches must be mDNS and SSDP enabled. All the edge switches must be configured to use the L2GRE tunnel of the Aruba wireless controller.

All the wireless traffic from Aruba APs is tunneled to the Aruba WLAN controller directly through the tunnel established between APs and Aruba WLAN controller.

The mDNS and SSDP traffic entering the edge switch is tunneled to the responder.

The mDNS and SSDP traffic received back from the controller on the L2GRE tunnel is verified. If the packet is unicast, it is forwarded based on the destination. If the packet is multicast, it is forwarded to the VLAN based on the VLAN ID tag in the packet.

The following diagram represents a sample Aruba mode setup:

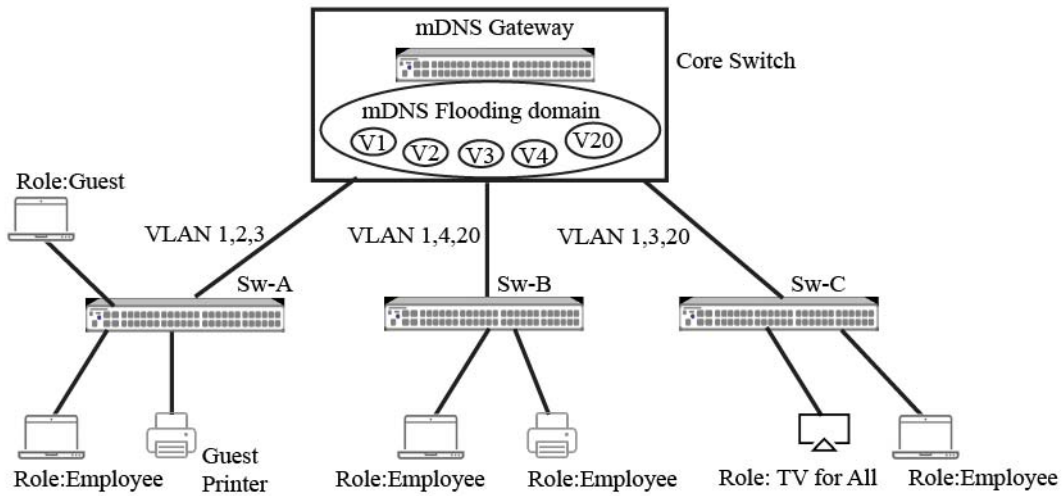


Gateway Mode

Gateway mode is configured if the network has no WLAN controller. In this mode, any one switch is configured as the mDNS and SSDP gateway. The traffic from the edge switches is forwarded to the configured gateway switch.

The gateway switch maintains the gateway VLAN list, which contains the VLANs to which the incoming mDNS and SSDP packets is flooded. Only mDNS and SSDP multicast packets are flooded to the gateway VLAN list. Unicast packets are not flooded to the gateway VLAN list.

The following diagram represents a sample Gateway mode setup:



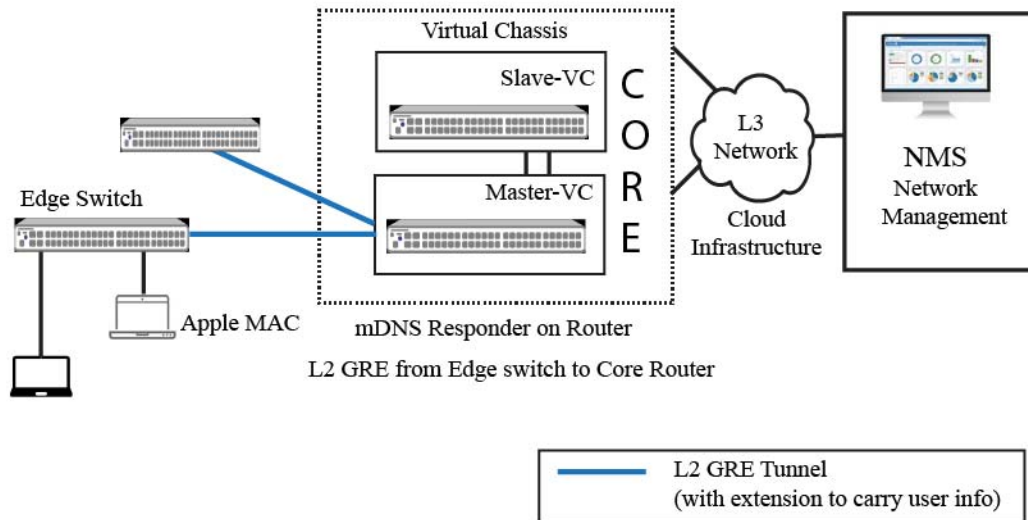
Tunnel Standard Mode

Tunnel Standard mode is configured if the network has controller other than Aruba controller. All the edge switches must be mDNS and SSDP enabled. All the edge switch are configured with the L2GRE tunnel with the remote tunnel endpoint IP address of the centralized OmniSwitch configured as responder.

The mDNS and SSDP traffic entering the edge switch is tunneled to the configured responder.

The mDNS and SSDP traffic received back from the controller on the L2GRE tunnel is verified. If the packet is unicast, it is forwarded based on the destination. If the packet is multicast, it is forwarded to the VLAN based on the VLAN ID tag in the packet (except the VLAN ID 4095). If the packet consists of VLAN tag 4095, the packet is flooded to the configured access VLAN list.

The following diagram represents a sample Standard mode setup:



Note. For configuration procedure of each mode, refer to [“Quick Steps for Zero Configuration” on page 35-78.](#)

Backward Compatibility

The configurations made using the older CLIs will continue to work provided the Loopback0 IP address is configured as the source endpoint address. Any further changes for the existing configuration can be made only after saving the configuration in the new format by executing the **write memory** command.

Verifying the Zero Configuration

To verify the Zero configuration on the switch, use the **show zeroconf config** command. The show command displays the configuration details with respect to the mode configured.

Example:

```
-> show zeroconf config
```

Example for default mode (Aruba mode)

```
zero-conf mode           : tunnel
zero-conf tunnel type    : aruba
gre-protocol             : 0x0
MDNS admin status       : disabled
SSDP admin status       : disabled
MDNS operational status  : down
SSDP operational status  : down
Responder IP            : -
Tunnel Source IP        : -
```

Example for standard mode

```
zero-conf mode           : tunnel
zero-conf tunnel type    : standard
gre-protocol             : 0x6558
MDNS admin status       : enabled
SSDP admin status       : enabled
MDNS operational status  : down
SSDP operational status  : down
Responder IP            : 10.0.0.1
Tunnel Source IP        : -
Access vlans list       : 1, 2, 3
```

Example for Gateway mode

```
zero-conf mode           : gateway
MDNS admin status       : enabled
SSDP admin status       : enabled
MDNS operational status  : up
SSDP operational status  : up
Gateway vlans list      : 4, 5, 6
```

Note. For more information on the CLI command usage, refer to *OmniSwitch AOS Release 6 CLI Reference Guide*.

BYOD Application Examples

The application scenarios provide various examples of how the ClearPass server and the OmniSwitch can be leveraged to provide different network access levels and UNPs for employees, guests, and other network-based devices.

In the following contexts, the main parameters like UNP name, VLAN number, and other parameters specified in the application examples are as follows:

Employee Registered Device - 802.1x Authentication

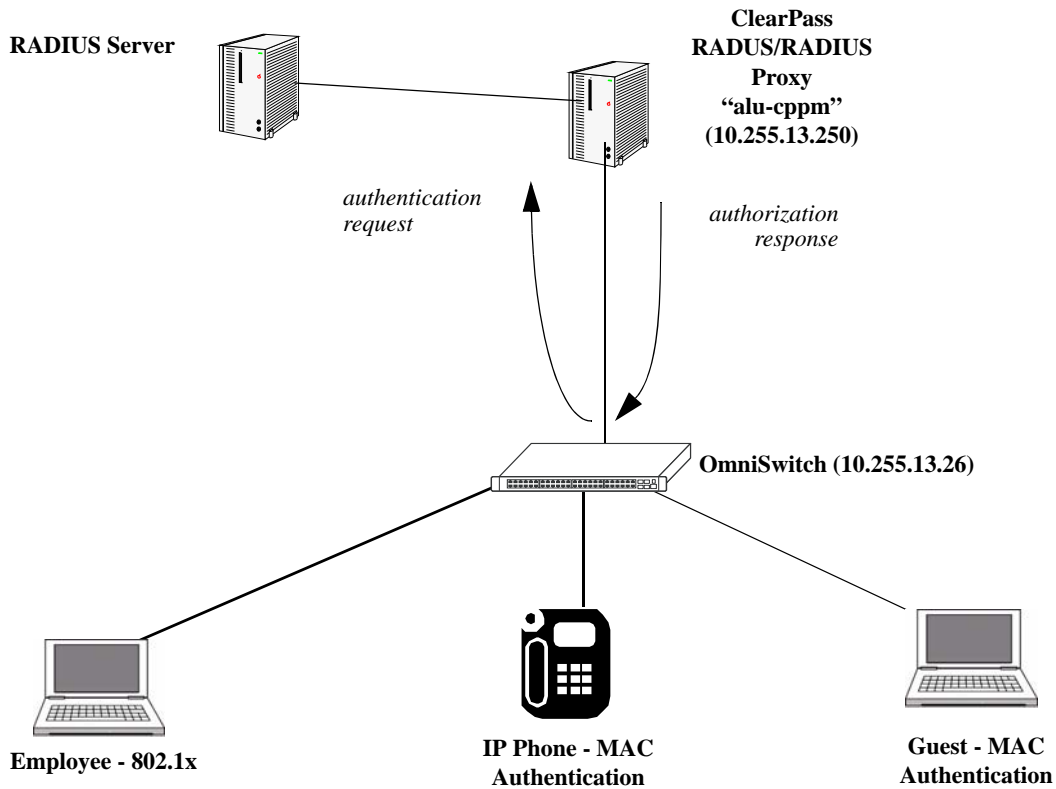
- UNP = UNP-employee
- VLAN = 96

IP Phone - MAC Authentication

- UNP = UNP-phone
- VLAN = 1002

Guest Device - MAC Authentication with Guest Login

- Registration UNP = UNP-Restricted
- Registration VLAN = 96
- Redirect Server = 10.255.95.206
- Guest UNP = UNP-guest
- Guest VLAN = 96



BYOD network with Employee and Guest devices

Application Example 1 (802.1x) - OmniSwitch Configuration

The OmniSwitch configuration for an 802.1x supplicant:

1 Configure 802.1x and port mobility as follows:

```
-> vlan port mobile 1/11
-> vlan port 1/11 802.1x enable
```

2 Configure 802.1x authentication for ClearPass RADIUS on an OmniSwitch as follows:

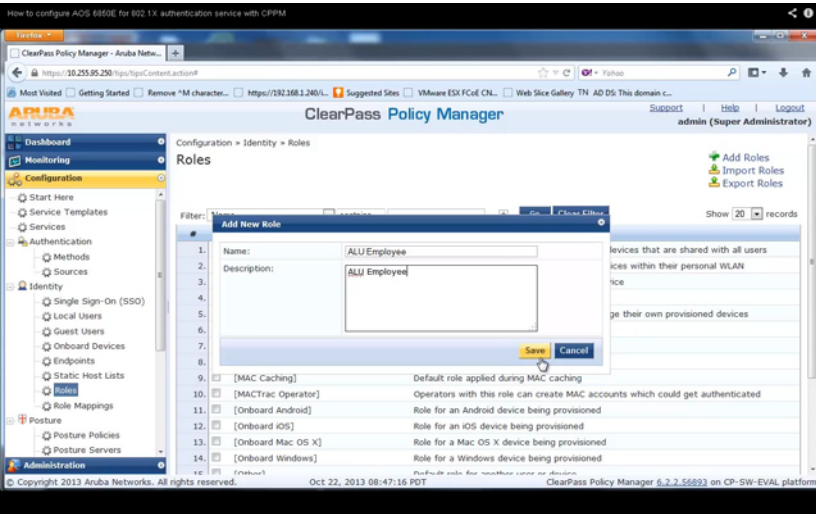
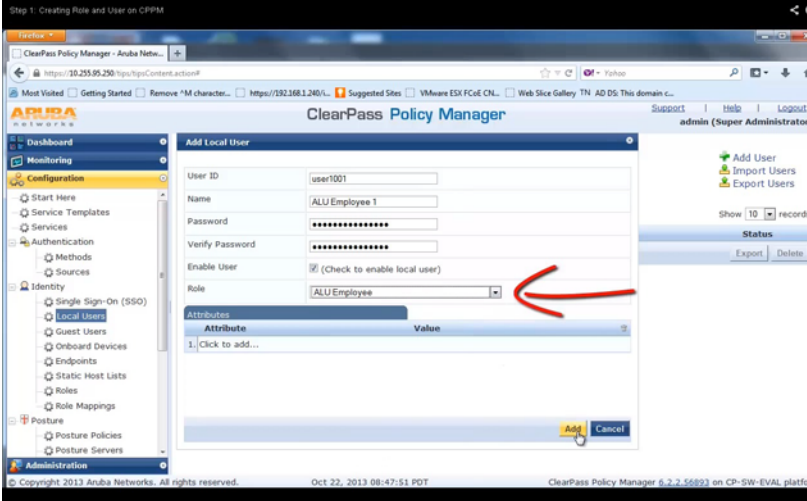
```
-> aaa radius-server alu-cppm host 10.255.95.250 key alcatel
-> aaa authentication 802.1x alu-cppm
```

3 Configure User Network Profiles as follows:

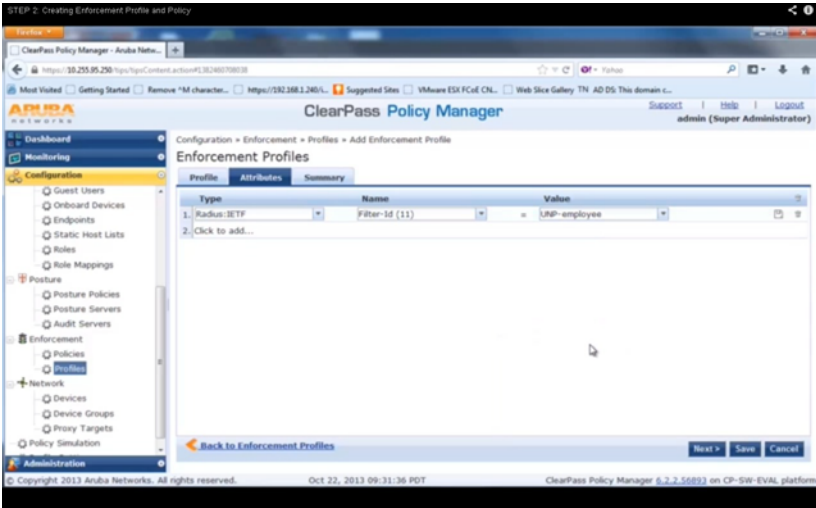
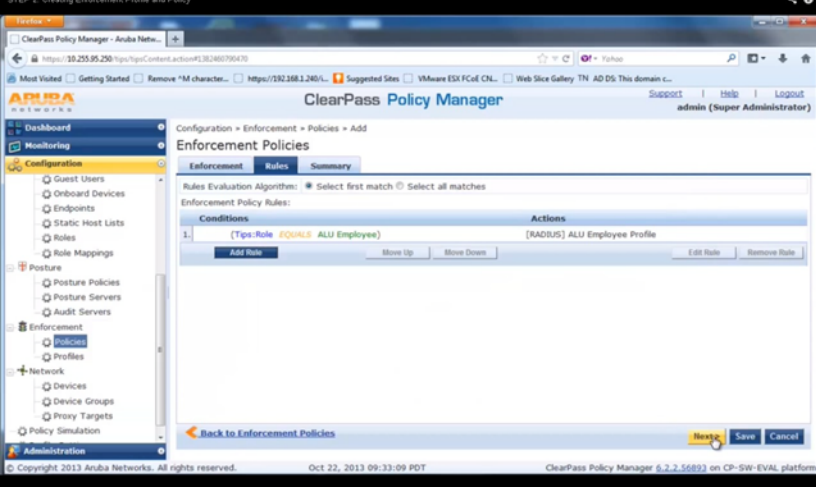
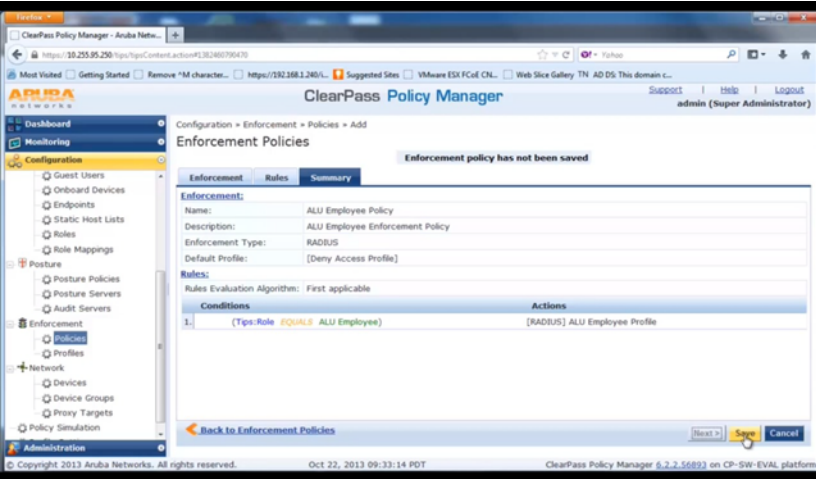
```
-> aaa user-network-profile name "UNP-employee" vlan 96
```

Application Example 1 (802.1x) - ClearPass Configuration

Step 1. ClearPass (802.1x) - Creating employee users and roles

| | |
|---|---|
| <p>Create user role: Roles->Add Roles</p> |  |
| <p>Create users and assign role: Local Users -> Add Users</p> |  |

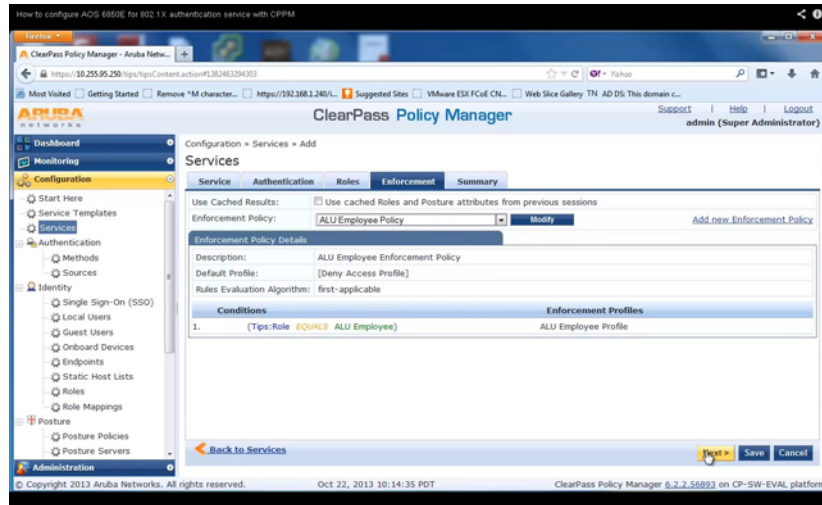
Step 2. ClearPass (802.1x) - Create Profiles and Policies

| | |
|---|--|
| <p>Create Profile:</p> <p>Attributes (tab)</p> <ul style="list-style-type: none"> - Type: Radius:IETF - Filter-ID (11) - Value = UNP-employee (Note: must match UNP Profile on OmniSwitch) |  |
| <p>Create Enforcement Policy:</p> <p>Rules (tab)</p> |  |
| <p>View Policies Summary</p> <p>Summary</p> |  |

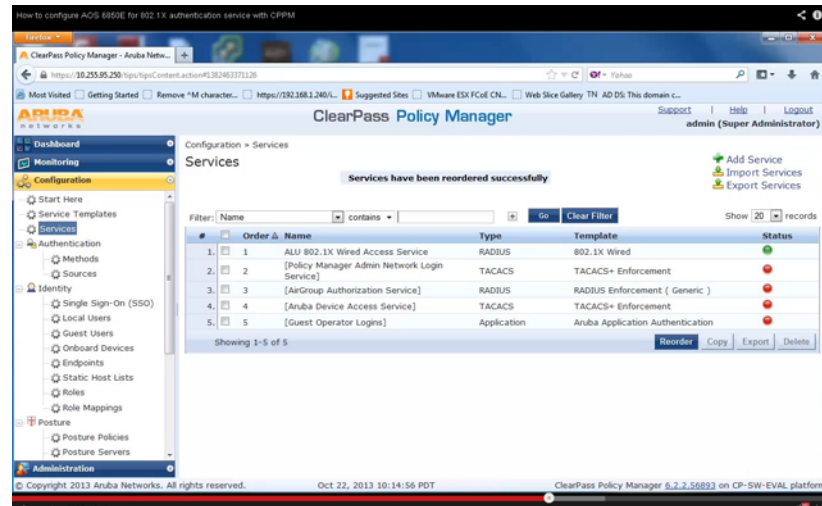
Step 3. ClearPass (802.1x) - Create 802.1X services

| <p>Add OmniSwitch to ClearPass Database</p> <p>Devices (tab)</p> | | | | | | | | | | | | | | | | | |
|---|--|------------|--|----------|-------|----------------|---------------|--------|---------------|----------------|--------------|------------|--|--------------------|--|--|--|
| <p>Add 802.1x Wired Service</p> <p>Service (tab)</p> | <table border="1"> <thead> <tr> <th>Type</th> <th>Name</th> <th>Operator</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>1. Radius:IETF</td> <td>NAS-Port-Type</td> <td>EQUALS</td> <td>Ethernet (15)</td> </tr> <tr> <td>2. Radius:IETF</td> <td>Service-Type</td> <td>BELONGS_TO</td> <td>Login-User (1), Framed-User (2), Authenticate-Only (8)</td> </tr> <tr> <td colspan="4">3. Click to add...</td> </tr> </tbody> </table> | Type | Name | Operator | Value | 1. Radius:IETF | NAS-Port-Type | EQUALS | Ethernet (15) | 2. Radius:IETF | Service-Type | BELONGS_TO | Login-User (1), Framed-User (2), Authenticate-Only (8) | 3. Click to add... | | | |
| Type | Name | Operator | Value | | | | | | | | | | | | | | |
| 1. Radius:IETF | NAS-Port-Type | EQUALS | Ethernet (15) | | | | | | | | | | | | | | |
| 2. Radius:IETF | Service-Type | BELONGS_TO | Login-User (1), Framed-User (2), Authenticate-Only (8) | | | | | | | | | | | | | | |
| 3. Click to add... | | | | | | | | | | | | | | | | | |
| <p>Configure 802.1x Authentication</p> <p>Authentication (tab)</p> | | | | | | | | | | | | | | | | | |

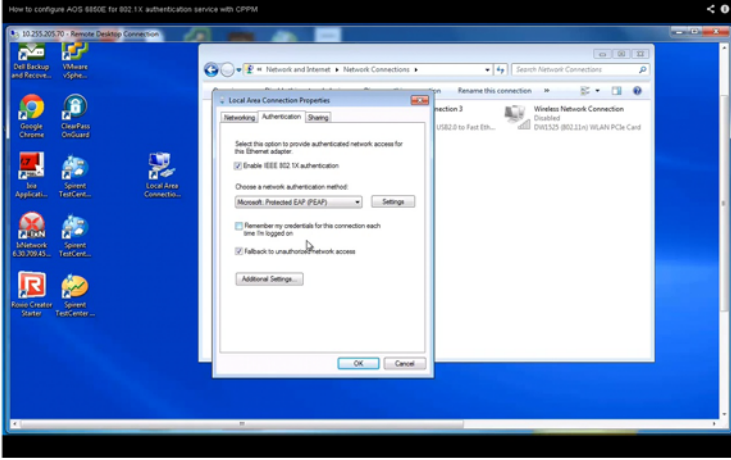
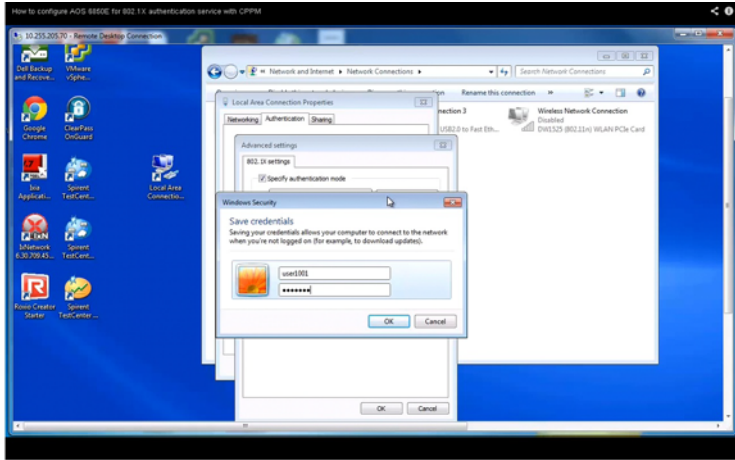
Configure Enforcement
Enforcement (tab)



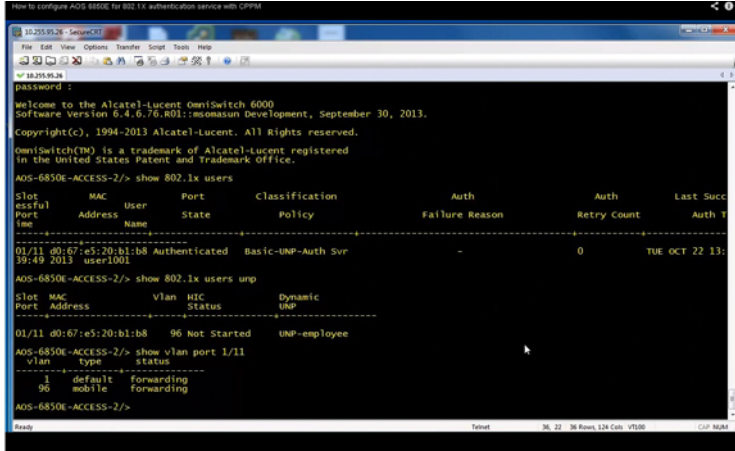
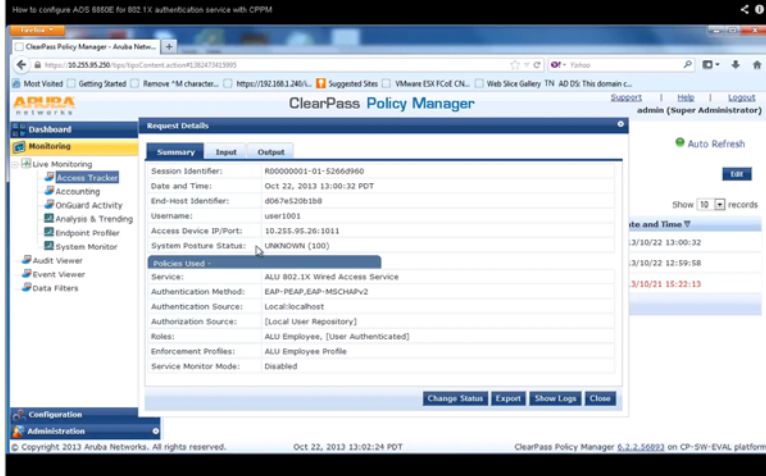
Reorder Authentication
Devices



Step 4. ClearPass (802.1x) - Configure PC

| | |
|--|---|
| <p>Configure PC Properties</p> |  <p>The screenshot shows a Windows desktop with a blue background. A window titled 'Local Area Connection Properties' is open, displaying the 'Authentication' tab. The 'Enable IEEE 802.1X authentication' checkbox is checked. Below it, 'Choose a network authentication method' is set to 'Microsoft Protected EAP (PEAP)'. There are also checkboxes for 'Remember my credentials for the connection each time I'm logged on' (unchecked) and 'Fallback to unauthenticated network access' (checked). An 'Additional Settings...' button is visible at the bottom.</p> |
| <p>Configure PC Advanced Settings</p> |  <p>The screenshot shows the same Windows desktop. The 'Local Area Connection Properties' window is still open, but the 'Advanced settings' dialog is now open over it. In the 'Advanced settings' dialog, the '802.1x settings' section is expanded, and the 'Specify authentication mode' checkbox is checked. A 'Windows Security' dialog is also open, titled 'Save credentials', with a message: 'Saving your credentials allows your computer to connect to the network when you're not logged on (for example, to download updates)'. It has a text field containing 'user1001' and a password field with masked characters. 'OK' and 'Cancel' buttons are at the bottom.</p> |

Step 5. ClearPass (802.1x) - Confirm Device Authentication

| <p>Confirm device Authentication</p> <p>OmniSwitch</p> |  <pre> AOS-6850E-ACCESS-2/ show 802.1x users ----- Slot MAC User Port Classification Auth Auth Last Succ Line Address Name State Policy Failure Reason Retry Count Auth T ----- 01/11 40:67:e5:20:b1:b8 user1001 Authenticated Basic-UNP-Auth Svr - 0 TUE OCT 22 13: 19:49 2013 user1001 AOS-6850E-ACCESS-2/ show 802.1x users urp ----- Slot MAC vlan HIC Dynamic Line Address Status ----- 01/11 40:67:e5:20:b1:b8 96 Not Started UNP-employee AOS-6850E-ACCESS-2/ show vlan port 1/11 ----- vlan type status ----- 1 default Forwarding 96 mobile Forwarding AOS-6850E-ACCESS-2/ </pre> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---------|-------|--------|---------------------|-----------------------|--|----------------|---------------------------|--|----------------------|--------------|--|-----------|----------|--|------------------------|-------------------|--|------------------------|---------------|--|----------------------|--|--|----------|---------------------------------|--|------------------------|-----------------------|--|------------------------|-----------------|--|-----------------------|-------------------------|--|--------|------------------------------------|--|-----------------------|----------------------|--|-----------------------|----------|--|
| <p>Confirm Device Authentication</p> <p>ClearPass</p> |  <p>Request Details</p> <table border="1"> <thead> <tr> <th>Summary</th> <th>Input</th> <th>Output</th> </tr> </thead> <tbody> <tr> <td>Session Identifier:</td> <td>80000001-01-5266-0960</td> <td></td> </tr> <tr> <td>Date and Time:</td> <td>Oct 22, 2013 13:00:32 PDT</td> <td></td> </tr> <tr> <td>End-Host Identifier:</td> <td>4067e520b1b8</td> <td></td> </tr> <tr> <td>Username:</td> <td>user1001</td> <td></td> </tr> <tr> <td>Access Device IP/Port:</td> <td>10.255.95.26:1011</td> <td></td> </tr> <tr> <td>System Posture Status:</td> <td>UNKNOWN (100)</td> <td></td> </tr> <tr> <td colspan="3">Policies Used</td> </tr> <tr> <td>Service:</td> <td>ALL 802.1X Wired Access Service</td> <td></td> </tr> <tr> <td>Authentication Method:</td> <td>EAP-PEAP,EAP-MSCHAPv2</td> <td></td> </tr> <tr> <td>Authentication Source:</td> <td>Local:localhost</td> <td></td> </tr> <tr> <td>Authorization Source:</td> <td>[Local User Repository]</td> <td></td> </tr> <tr> <td>Roles:</td> <td>ALL Employee, [User Authenticated]</td> <td></td> </tr> <tr> <td>Enforcement Profiles:</td> <td>ALL Employee Profile</td> <td></td> </tr> <tr> <td>Service Monitor Mode:</td> <td>Disabled</td> <td></td> </tr> </tbody> </table> | Summary | Input | Output | Session Identifier: | 80000001-01-5266-0960 | | Date and Time: | Oct 22, 2013 13:00:32 PDT | | End-Host Identifier: | 4067e520b1b8 | | Username: | user1001 | | Access Device IP/Port: | 10.255.95.26:1011 | | System Posture Status: | UNKNOWN (100) | | Policies Used | | | Service: | ALL 802.1X Wired Access Service | | Authentication Method: | EAP-PEAP,EAP-MSCHAPv2 | | Authentication Source: | Local:localhost | | Authorization Source: | [Local User Repository] | | Roles: | ALL Employee, [User Authenticated] | | Enforcement Profiles: | ALL Employee Profile | | Service Monitor Mode: | Disabled | |
| Summary | Input | Output | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Session Identifier: | 80000001-01-5266-0960 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Date and Time: | Oct 22, 2013 13:00:32 PDT | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| End-Host Identifier: | 4067e520b1b8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Username: | user1001 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Access Device IP/Port: | 10.255.95.26:1011 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| System Posture Status: | UNKNOWN (100) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Policies Used | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Service: | ALL 802.1X Wired Access Service | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Authentication Method: | EAP-PEAP,EAP-MSCHAPv2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Authentication Source: | Local:localhost | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Authorization Source: | [Local User Repository] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Roles: | ALL Employee, [User Authenticated] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Enforcement Profiles: | ALL Employee Profile | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Service Monitor Mode: | Disabled | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Application Example 2 (IP Phone) - OmniSwitch Configuration

The OmniSwitch configuration for a non-suppliant IP phone:

1 Configure 802.1x and port mobility as follows:

```
-> vlan port mobile 1/13
```

```
-> vlan port 1/13 802.1x enable
```

```
-> 802.1x 1/13 non-suppliant policy authentication pass default-vlan fail block
```

2 Configure MAC-authentication for ClearPass RADIUS on an OmniSwitch as follows:

```
-> aaa radius-server alu-cppm host 10.255.95.250 key alcatel
```

```
-> aaa authentication mac alu-cppm
```

3 Configure User Network Profiles as follows:

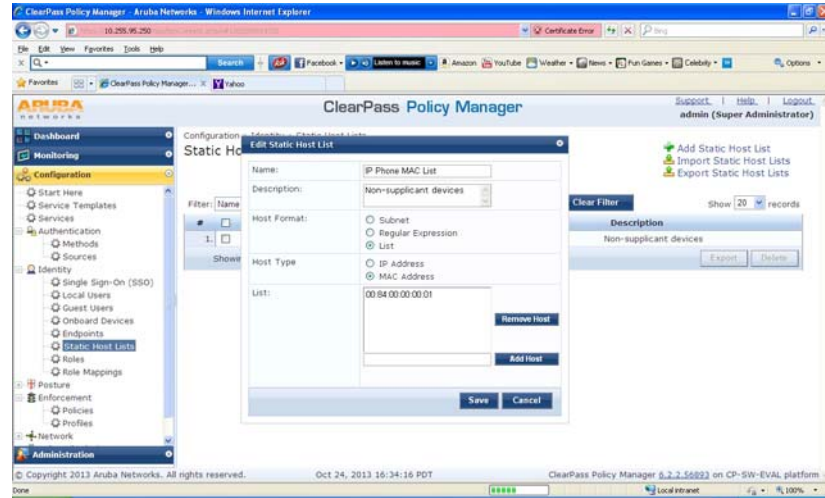
```
-> aaa user-network-profile name "UNP-phone" vlan 1002
```


Application Example 2 (IP Phone) - ClearPass Configuration

Step 1. ClearPass (IP Phone) - Creating static host list

Create static host list:

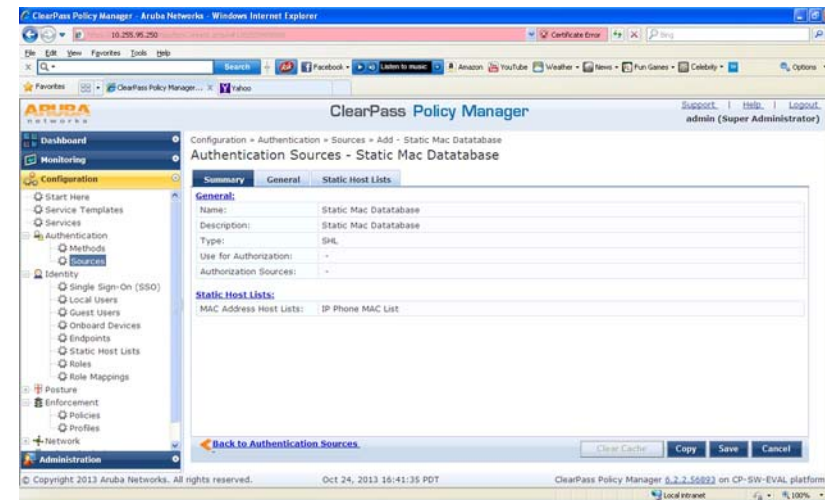
Identity->Static Host List



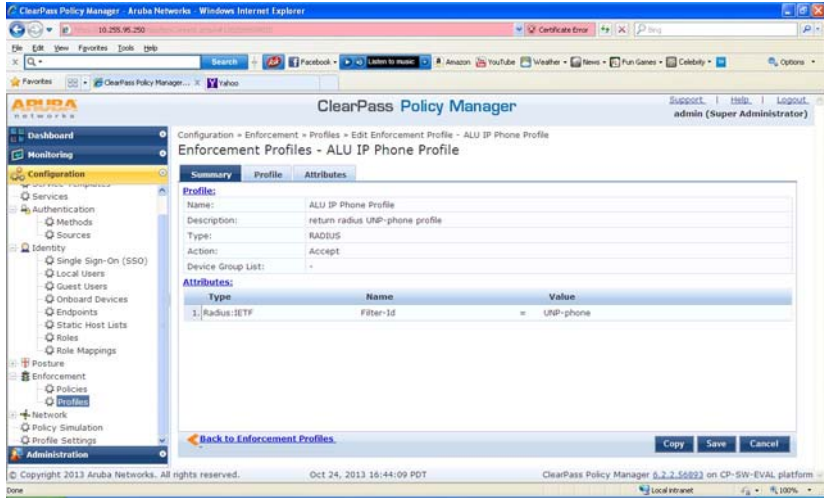
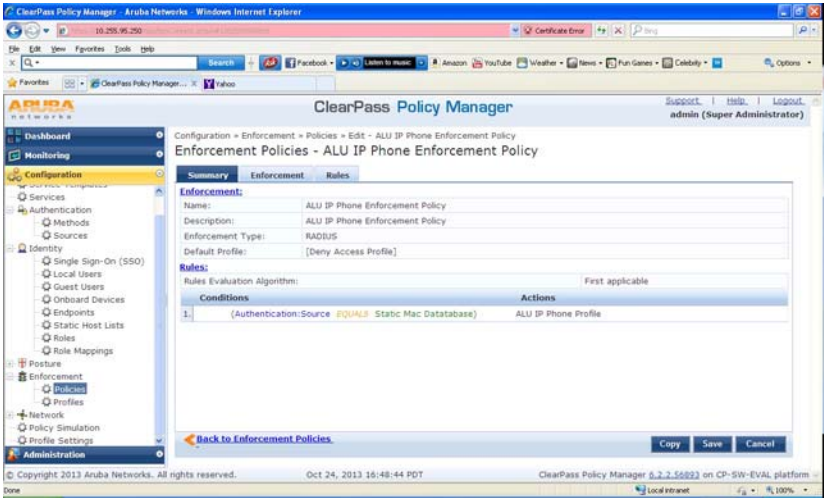
Create Authentication Source

Authentication-Sources-Add Authentication Source

Type: Static Host List
Host List: IP Phone MAC List



Step 2. ClearPass (IP Phone) - Create Profiles and Policies

| <p>Create Profile:</p> <p>Profile (tab)</p> <ul style="list-style-type: none"> - Name: ALU IP Phone Profile - Template: Aruba RADIUS Enforcement <p>Attributes (tab)</p> <ul style="list-style-type: none"> - Type: Radius:IETF - Filter-ID (11) - Value = UNP-phone (Note: must match UNP Profile on OmniSwitch) |  <p>The screenshot shows the 'ClearPass Policy Manager' interface. The breadcrumb trail is 'Configuration > Enforcement > Profiles > Edit Enforcement Profile - ALU IP Phone Profile'. The 'Attributes' tab is selected, showing a table with one attribute:</p> <table border="1"> <thead> <tr> <th>Type</th> <th>Name</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Radius:IETF</td> <td>Filter-Id</td> <td>= UNP-phone</td> </tr> </tbody> </table> | Type | Name | Value | Radius:IETF | Filter-Id | = UNP-phone |
|---|---|-------------|---------|---|----------------------|-----------|-------------|
| Type | Name | Value | | | | | |
| Radius:IETF | Filter-Id | = UNP-phone | | | | | |
| <p>Create Enforcement Policy:</p> <p>Rules (tab)</p> <ul style="list-style-type: none"> - Type: Authentication - Name: Source - Operator: EQUALS - Value: Static Mac Database <p>Profile Name: ALU IP Phone Profile</p> |  <p>The screenshot shows the 'ClearPass Policy Manager' interface. The breadcrumb trail is 'Configuration > Enforcement > Policies > Edit - ALU IP Phone Enforcement Policy'. The 'Rules' tab is selected, showing the following configuration:</p> <p>Enforcement:</p> <ul style="list-style-type: none"> Name: ALU IP Phone Enforcement Policy Description: ALU IP Phone Enforcement Policy Enforcement Type: RADIUS Default Profile: [Deny Access Profile] <p>Rules:</p> <p>Rules Evaluation Algorithm: First applicable</p> <table border="1"> <thead> <tr> <th>Conditions</th> <th>Actions</th> </tr> </thead> <tbody> <tr> <td>1. (Authentication:Source EQUALS Static Mac Database)</td> <td>ALU IP Phone Profile</td> </tr> </tbody> </table> | Conditions | Actions | 1. (Authentication:Source EQUALS Static Mac Database) | ALU IP Phone Profile | | |
| Conditions | Actions | | | | | | |
| 1. (Authentication:Source EQUALS Static Mac Database) | ALU IP Phone Profile | | | | | | |

Step 3. ClearPass (IP Phone) - Create MAC Authentication Service

Add MAC Authentication Service

Service (tab)
-Type: MAC Authentication

Authentication (tab)
- **Authentication**
Sources: Static MAC Database

Enforcement (tab)
- **Enforcement Policy:**
ALU IP Phone Enforcement Policy

The screenshot displays the ClearPass Policy Manager interface. The main content area shows the configuration for an 'ALU MAC-based Authentication Service'. The 'Service' tab is selected, showing the following details:

- Name:** ALU MAC-based Authentication Service
- Description:** ALU MAC-based Authentication Service
- Type:** MAC Authentication
- Status:** Enabled
- Monitor Mode:** Disabled

Below the service details, the 'Service Rule' section is visible, indicating that the rule matches all of the following conditions:

| Type | Name | Operator | Value |
|----------------|---------------|------------|---------------------------------|
| 1. Radius:RTPF | NAS-Port-Type | BELONGS_TO | Ethernet [15] |
| 2. Radius:RTPF | Service-Type | BELONGS_TO | Login-User (1), Call-Check (10) |

The 'Authentication' section shows 'Authentication Methods' set to '[Allow All MAC AUTH]' and 'Authentication Sources' set to 'Static Mac Database'. Buttons for 'Disable', 'Copy', 'Save', and 'Cancel' are located at the bottom right of the configuration area.

Application Example 3 (Guest) - OmniSwitch Configuration

The OmniSwitch configuration for guest UNP, VLANs, and redirection:

1 Configure 802.1x and port mobility as follows:

```
-> vlan port mobile 1/13
-> vlan port 1/13 802.1x enable
```

2 Configure MAC-authentication for ClearPass RADIUS on an OmniSwitch as follows:

```
-> aaa radius-server alu-cppm host 10.255.95.250 key alcatel
-> aaa authentication 802.1x alu-cppm
-> aaa authentication mac alu-cppm
-> aaa accounting 802.1x alu-cppm
-> aaa accounting mac alu-cppm
```

3 Configure User Network Profiles and redirect server as follows:

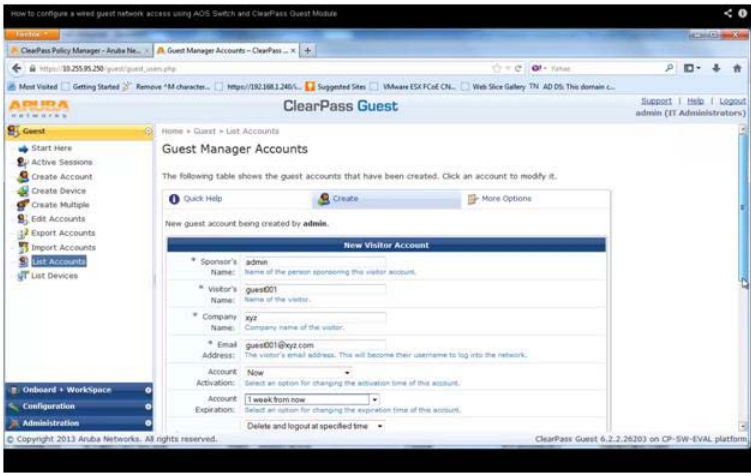
```
-> aaa user-network-profile name "UNP-guest" vlan 96
-> aaa user-network-profile name "UNP-restricted" vlan 96
-> aaa redirect-server alu-cppm ip-address 10.255.95.250
```

Application Example 3 (Guest) - ClearPass Configuration

Step 1. ClearPass (Guest) - Create Guest Account and Web login page

Create guest account

Guest->List Accounts



The screenshot shows the ClearPass Guest web interface. The browser address bar indicates the URL is https://192.168.1.240/guest/guest_admin.php. The page title is 'ClearPass Guest'. The main content area is titled 'Guest Manager Accounts' and displays a 'New Visitor Account' form. The form fields include: Sponsor's Name (admin), Visitor's Name (guest001), Company Name (xyz), Email (guest001@xyz.com), Account (Now), Activation (Select an option for changing the activation time of this account), and Expiration (1 week from now). There are 'Create' and 'More Options' buttons at the top of the form. The footer of the page reads '© Copyright 2013 Aruba Networks. All rights reserved. ClearPass Guest 6.2.2.26203 on CP-SW-EVAL platform'.

Create web login page

Configuration-Web Logins

Name- Alcatel-Lucent Secure Access

Page name: secure-access

Vendor Settings: Alcatel-Lucent

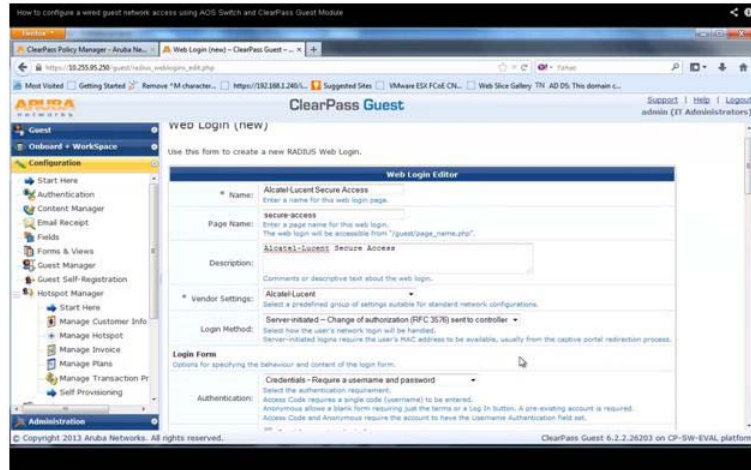
Login Method: Server-initiated

Pre-Auth Check:None

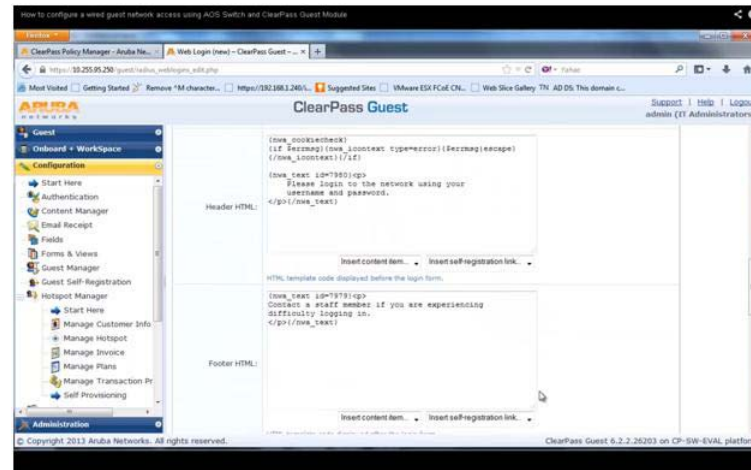
Terms: checked

Default URL: www.google.com

Override Destination: checked

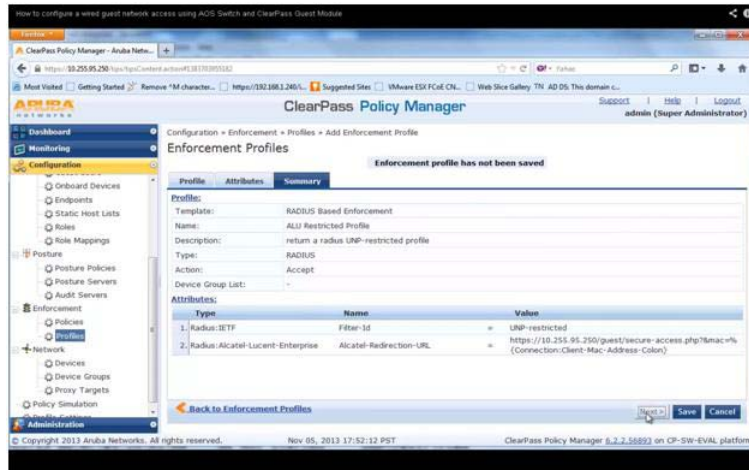


Create custom skin if desired

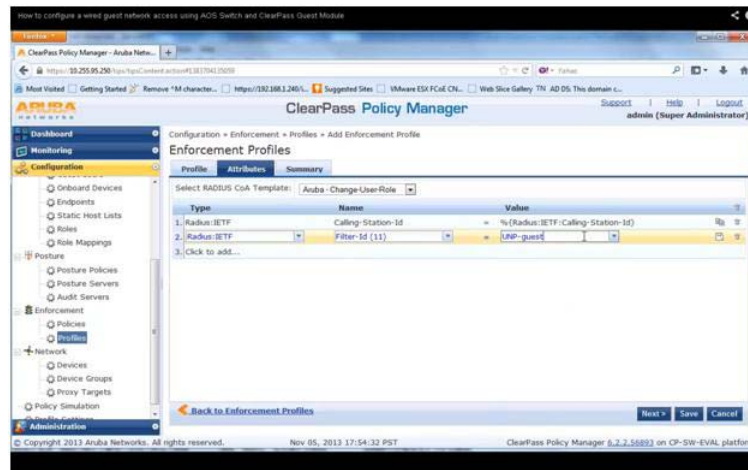


Step 2 ClearPass (Guest) - Create Profiles

Create Restricted Profile:
Enforcement->Profiles
Template: RADIUS Based Enforcement
Name: ALU Restricted Profile
Type: RADIUS
Action: Accept
Attribute Type: Radius:IETF, Alcatel-Lucent-Enterprise
Attribute Name: Filter-ID, Alcatel-Redirection-URL
Attribute Value: UNP-restricted, (redirect URL)



Create Guest Profile:
Enforcement->Profiles
Template: RADIUS Change of Authorization (CoA)
Name: ALU Guest CoA Profile
RADIUS CoA Template: Aruba-Change-User-Role
Attributes Type: Radius:IETF
Attribute Name: Filter-ID
Attribute Value: UNP-guest

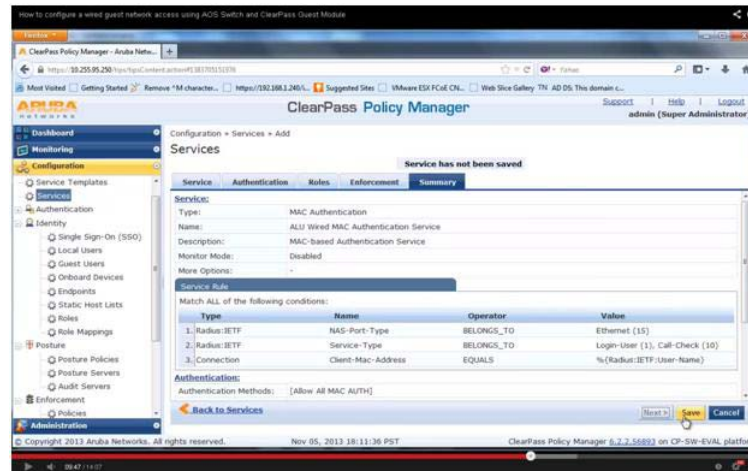


Step 3 ClearPass (Guest) - Create MAC and Web Authentication Services

Add MAC
Authentication Service

Configuration->Services

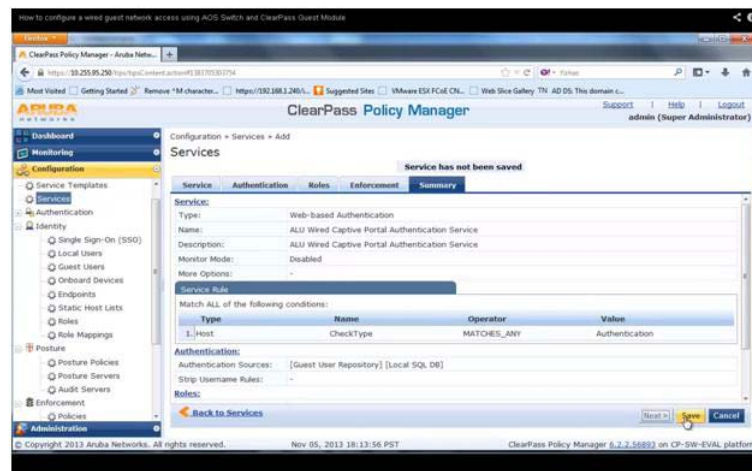
Type: MAC
Authentication
Name: ALU Wired
MAC Authentication
Service
Monitor Mode:
Disabled
Service Rule Type:
Radius:IETF
Service Rule Name:
NAS-Port-Type
**Service Rule
Operator:**
BELONGS_TO
Service Rule Value:
Ethernet (15)
**Authentication
Methods:** Allow All
MAC AUTH
Enforcement Policy:
ALU Wired MAC
Enforcement Policy



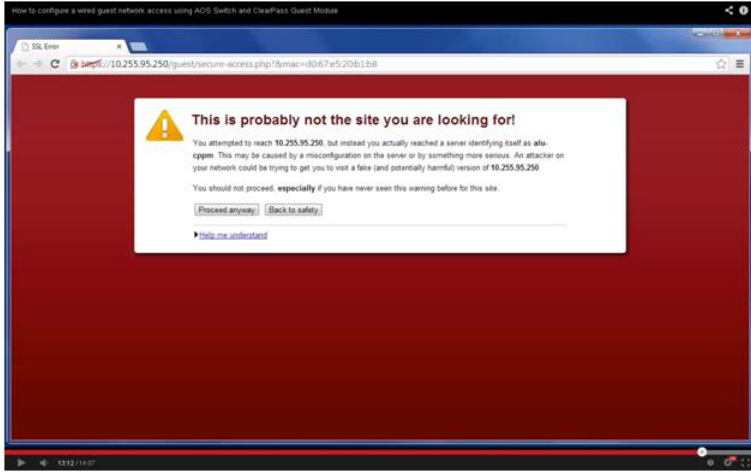
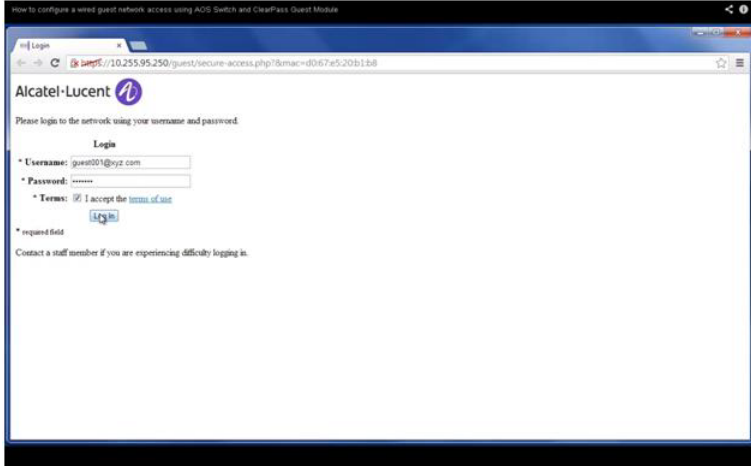
Add Web
Authentication Service

Configuration->Services

Type: Web-based
Authentication
Name: ALU Wired
Captive Portal
Authentication Service
**Authentication
Sources:** [Guest user
Repository] [Local SQL
DB]
Enforcement Policy:
ALU Wired Captive
Portal Enforcement
Policy



Step 3 ClearPass (Guest) - Login Example

| | |
|-------------------------|--|
| <p>Example Redirect</p> |  <p>The screenshot shows a web browser window with a red background and a white warning box. The warning text reads: "This is probably not the site you are looking for! You attempted to reach 10.255.95.250, but instead you actually reached a server identifying itself as alacppm. This may be caused by a misconfiguration on the server or by something more serious. An attacker on your network could be trying to get you to visit a fake (and potentially harmful) version of 10.255.95.250. You should not proceed, especially if you have never seen this warning before for this site." Below the text are buttons for "Proceed anyway" and "Back to safety", and a link for "help me understand".</p> |
| <p>Example login</p> |  <p>The screenshot shows a web browser window displaying the Alcatel-Lucent login page. The page title is "Login" and the Alcatel-Lucent logo is visible. The text says "Please login to the network using your username and password." Below this is a "Login" section with a "Username" field containing "guest001@xyz.com", a "Password" field with masked characters, and a "Terms" section with a checked checkbox and a link to "terms of use". There is also a "Login" button and a note: "Contact a staff member if you are experiencing difficulty logging in."</p> |

Verifying BYOD Configuration

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

| | |
|--------------------------------------|---|
| show aaa redirect-server | Displays redirection server name and its details. |
| show aaa port-bounce status | Displays the status of global and port specific port bounce configuration. |
| show aaa redirect pause-timer | Displays the configured global pause-timer value. |
| show byod host | Displays the status of the new BYOD clients that come to the network. |
| show byod status | Displays the status of the new client that enters the network at the particular port. |

36 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-ACE/Server.

In This Chapter

The chapter includes 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 36-9.
- **Configuring a RADIUS Server.** This procedure is described in [“RADIUS Servers”](#) on page 36-10.
- **Configuring a TACACS+ Server.** This procedure is described in [“TACACS+ Server”](#) on page 36-31.
- **Configuring an LDAP Server.** This procedure is described in [“LDAP Servers”](#) on page 36-34.

For information about using servers for authenticating users to manage the switch, see the “Switch Security” chapter in the *OmniSwitch AOS Release 6 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 through 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 6350, 6450 |
| 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 AOS Release 6 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 following 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 | retransmit | 3 |
| Timeout for server replies to authentication requests | timeout | 2 |
| UDP destination port for authentication | auth-port | 1645* |
| UDP destination port for accounting | acct-port | 1646* |
| MAC address format status for authentication | mac-address-format-status | disable |
| MAC address format for authentication | mac-address-format | uppercase |
| Physical port of the NAS server. | nas-port | default |
| The interface identifier of the NAS port authenticating the user | nas-port-id | disable |
| Type of the physical port of the NAS server that is authenticating the user | nas-port-type | Ethernet |
| Unique session ID for RADIUS accounting | unique-acct-session-id | disable |

* 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 | timeout | 2 |
| The port number for the server | port | 49 |

LDAP Authentication Servers

Defaults for the `aaa ldap-server` command are as follows:

| Description | Keyword | Default |
|---|---------------------|---|
| The port number for the server | port | 389 (SSL disabled) 636 (SSL enabled) |
| Number of retries on the server before the switch tries a backup server | retransmit | 3 |
| Timeout for server replies to authentication requests | timeout | 2 |
| Whether a Secure Socket Layer is configured for the server | ssl no ssl | no ssl |

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 36-10, “TACACS+ Server” on page 36-31, and “LDAP Servers” on page 36-34.
- 2 Use the **aaa radius-server**, **aaa tacacs+-server**, and/or **aaa ldap-server** commands to configure the authentication servers. 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
  Nas port               = default,
  Nas port id           = disable,
  Nas port type         = ethernet,
  MAC Address Format Status= enable,
  MAC Address Format     = uppercase,
  Unique Acct Session Id = disable

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 *OmniSwitch AOS Release 6 CLI Reference Guide* for information about the fields in this display.

- 3 If you are using ACE/Server, switch configuration is not required; however, FTP the **sdconf.rec** file from the server to the **/network** directory of the switch.
- 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 AOS Release 6 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-ACE/Server can 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 can 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 can 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) can specify a list of backup authentication servers that includes servers of different types (if supported on the feature).

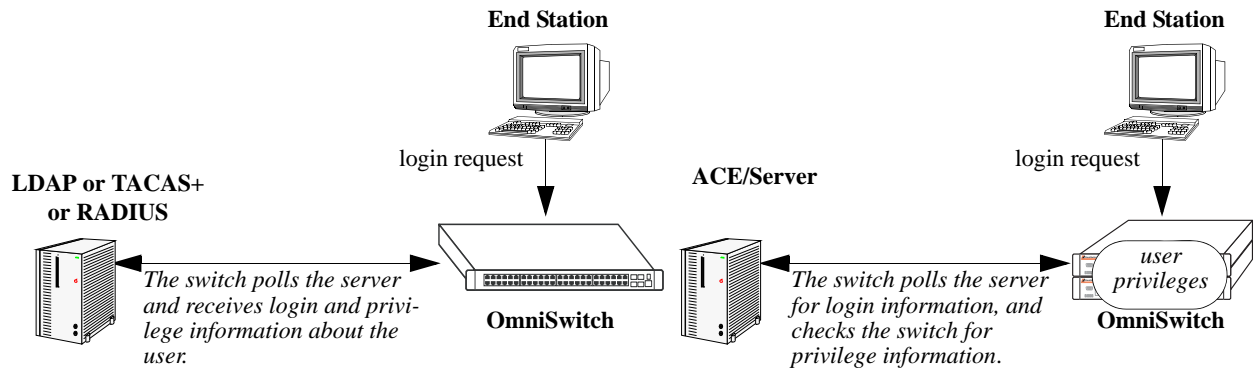
The switch uses the first available authentication server to attempt to authenticate users. If user information is not found on the first available server, the authentication attempts fails.

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.

Additional servers can be configured as backups for RADIUS, TACACS+, and LDAP servers.

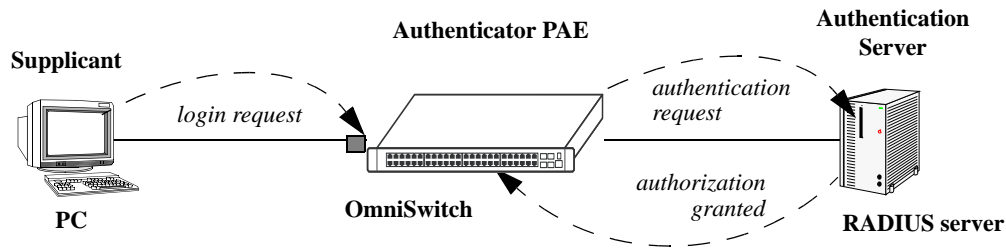
A RADIUS server supporting the challenge and response mechanism as defined in RADIUS RFC 2865 can access an ACE/Server for authentication purposes. The ACE/Server is then used for user authentication, and the RADIUS server is used for user authorization.



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 can also contain challenges or 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 can 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 AOS Release 6 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 in to 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 privileges of the user. 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 AOS Release 6 Switch Management Guide*.

Server-specific parameter configurations are not required for the switch to communicate with an attached ACE/Server; however, FTP the **sdconf.rec** file from the server to the **/network** directory of the switch. **sdconf.rec** file is required so that the switch knows the IP address of the ACE/Server. For information about loading files on to the switch, see the *OmniSwitch AOS Release 6 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 can 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 servers configured for authentication, use the **show aaa server** command. For more information about the output for this command, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Also, clear the ACE/Server secret occasionally for misconfiguration or required changes in configuration.

Clearing an ACE/Server Secret

The ACE/Server generates “secrets” that is sent to clients for authentication. While you cannot configure the secret on the switch, you can clear it. The secret must 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, it is also required to clear the secret on the ACE/Server as described in 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 can include VLAN information, time-of-day, or slot/port restrictions.

RADIUS Server Attributes

RADIUS servers and RADIUS accounting servers are configured with specific 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 standard RADIUS server 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 36-13](#). Attributes 40–59 are used for RADIUS accounting servers and are listed in [“RADIUS Accounting Server Attributes” on page 36-14](#).

| Standard Attribute Number | Standard Attribute | Notes |
|---------------------------|--------------------|--|
| 1 | User-Name | Used in access-request and account-request packets. |
| 2 | User-Password | Used in access-request. |
| 3 | CHAP-Password | <i>Not supported.</i> |
| 4 | NAS-IP-Address | Sent with every access-request. Specifies the switches a user can 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 | Framed-User (2) if authentication request type is: <ul style="list-style-type: none"> - supplicant/802.1x authentication - captive-portal authentication - ASA authentication Call-Check (10) if authentication request type is: <ul style="list-style-type: none"> - MAC based authentication |
| 7 | Framed-IP-Address | The IP address of the successfully authenticated supplicant is input to the Framed-IP-Address attribute. |

| Standard Attribute Number | Standard Attribute | Notes |
|---------------------------|--------------------|---|
| 6 | Service-Type | Framed-User (2) if authentication request type is: - supplicant/802.1x authentication - captive-portal authentication - ASA authentication Call-Check (10) if authentication request type is: - MAC based authentication |
| 8 | Framed-Protocol | <i>Not supported. These attributes are used for dial-up sessions; not applicable to the RADIUS client in the switch.</i> |
| 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 | <i>Not supported.</i> |
| 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. |
| 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 36-13. |
| 27 | Session-Timeout | Supported |
| 28 | Idle-Timeout | <i>Not supported.</i> |
| 31 | Calling-Station-ID | See “Calling-Station-ID in RADIUS Access and Accounting Packets” on page 36-16. |

| Standard Attribute Number | Standard Attribute | Notes |
|----------------------------------|---------------------------------|--|
| 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 | |
| 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 | |

Client IP in Accounting Message

To include the user IP address in the accounting request packet of 802.1x authentication, 802.1x accounting server and DHCP snooping features must be enabled on the switch. Upon a successful 802.1x authentication the user IP address is sent to the accounting server in the Framed-IP-Address attribute of the accounting request packet. This is a default feature that cannot be disabled.

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 RADIUS server configurations of the vendors.

The attribute subtypes are defined in the dictionary file of the server. Alcatel-Lucent vendor ID is 800 (SMI Network Management Private Enterprise Code).

The following are VSAs for RADIUS servers:

| Number | RADIUS VSA | Type | Description |
|--------|-----------------------------|---------|--|
| 1 | Alcatel-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-Slot-Port | string | Slots/ports valid for the user. |
| 3 | Alcatel-Time-of-Day | string | The time of day valid for the user to authenticate. |
| 4 | Alcatel-Client-IP-Addr | address | The IP address used for Telnet only. |
| 5 | Alcatel-Group-Desc | string | Description of the VLAN. |
| 6 | Alcatel-Port-Desc | string | Description of the port. |
| 8 | Alcatel-Auth-Group-Protocol | string | The protocol associated with the VLAN. Configure this for access to other protocols. Values include: IP_E2 , IP_SNAP . |
| 9 | Alcatel-Asa-Access | string | Specifies that the user has access to the switch. The only valid value is all . |
| 39 | Alcatel-Acce-Priv-F-R1 | hex. | Configures functional read privileges for the user. |
| 40 | Alcatel-Acce-Priv-F-R2 | hex. | Configures functional read privileges for the user. |
| 41 | Alcatel-Acce-Priv-F-W1 | hex. | Configures functional write privileges for the user. |
| 42 | Alcatel-Acce-Priv-F-W2 | hex. | Configures functional write privileges for the user. |

The Alcatel-Auth-Group attribute is used for Ethernet II only. If a different protocol, or more than one protocol is required, use the Alcatel-Auth-Group-Protocol attribute instead. For example:

```
Alcatel-Auth-Group-Protocol 23: IP_E2 IP_SNAP
```

In this example, authenticated users on VLAN 23 can use Ethernet II or SNAP encapsulation.

Configuring Functional Privileges on the Server

Configuring the functional privileges attributes (**Alcatel-Acct-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 AOS Release 6 Switch Management Guide*.

RADIUS Accounting Server Attributes

The following table lists the standard attributes supported for RADIUS accounting servers. You can modify the attributes in the **radius.ini** file, if necessary.

| No. | Standard Attribute | Description |
|-----|---------------------|--|
| 1 | User-Name | Used in access-request and account-request packets. |
| 4 | NAS-IP-Address | Sent with every access-request. Specifies the switches a user can 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 must be included in the dictionary file: 1 (acct-start), 2 (acct-stop), 6 (failure), and 7 (acct-on). Start and stop correspond to login or 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 must 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. |

| No. | Standard Attribute | Description |
|-----|------------------------------|---|
| 49 | Acct-Terminal-Cause | Indicates how the session was terminated: NAS-ERROR USER-ERROR LOST CARRIER USER-REQUEST STATUS-FAIL |
| 52 | Acct-Input-Gigawords | Indicates the number of times Acct-Input-Octets counter has wrapped the 2 ³² (4GB) traffic over the course of the service being provided. This attribute is present in Accounting-Request records where the Acct-Status-Type is set to 'Stop' or 'Interim-Update'. |
| 53 | Acct-Output-Gigawords | Indicates the number of times Acct-Output-Octets counter has wrapped the 2 ³² (4GB) traffic in the course of delivering the service. This attribute is present in Accounting-Request records where the Acct-Status-Type is set to 'Stop' or 'Interim-Update'. |

The following table lists the VSAs supported for RADIUS accounting servers. You can modify the attributes in the **radius.ini** file, if necessary.

| No. | Accounting VSA | Type | Description |
|-----|-------------------------------|-----------------------|--|
| 1 | Alcatel-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-Slot-Port | string | Slots/ports valid for the user. |
| 4 | Alcatel-Client-IP-Addr | dotted decimal | The IP address used for Telnet only. |
| 5 | Alcatel-Group-Desc | string | Description of the VLAN. |

Calling-Station-ID in RADIUS Access and Accounting Packets

Calling-Station-ID is used by the NAS (Network Access Server) in an Access-Request packet to indicate that a call is being received. Based on the Calling-Station-ID or Calling-Station-ID attribute, the RADIUS server sends an Access-Accept to answer the call or an Access-Reject to reject the call.

Calling-Station-ID attribute is supported in Access-Request and Accounting-Request packet for Authenticated Switch Access users (ASA) users, supplicant and non-suppliant users, and captive portal users.

Note.

- Authentication and accounting server must be configured as RADIUS for 802.1x supplicant clients, non-suppliant clients, ASA users, and captive portal users. For more information on configuring authentication and accounting server, see chapter “AAA Commands” in the *OmniSwitch AOS Release 6 CLI Reference Guide*.
 - This feature is not supported when the authentication server and accounting server is configured as LDAP server, TACACS+ server, or ACE server.
-

The following table shows the default behavior of Calling-Station-ID attribute in RADIUS packets:

| RADIUS Message | ASA users | Suppliant users | Non-Suppliant users | Captive Portal users |
|---------------------------|---|---|--|--|
| Access-Request | Not applicable | Access request is sent with Calling-Station-ID (MAC address of the supplicant user) connecting to the OmniSwitch. | Access request is sent with Calling-Station-ID (MAC address of the non-suppliant user) connecting to the OmniSwitch. | Not applicable |
| Accounting-Request | Account request message is sent with Calling-Station-ID (IP address of the ASA user/host) connecting to the OmniSwitch. | Accounting request is sent with Calling-Station-ID (MAC address of the supplicant user) connecting to the OmniSwitch. | Accounting request is sent with Calling-Station-ID (MAC address of the non-suppliant user) connecting to the OmniSwitch. | Accounting request is sent with Calling-Station-ID (IP address of the captive portal user) connecting to the OmniSwitch. |

NAS-Identifier Support in RADIUS Access Request Packets

Configure the NAS-Identifier, which specifies originating NAS device sending access request frame using the NAS identifier string.

To configure the NAS-Identifier, use the **aaa radius nas-identifier** command. If the NAS-Identifier is configured as **default**, then the system name would be sent in the access-request frames. If the NAS-Identifier is configured using the user-string, then the user-defined NAS-Identifier will be used.

By default, the value of NAS-Identifier is 'default'.

For example,

```
-> aaa radius nas-identifier user-string "hello"
-> aaa radius nas-identifier default
```

Use **show aaa radius config** and **aaa configuration snapshot** to view the NAS-Identifier configuration. 'aaa configuration snapshot' command will display the value of NAS-Identifier only when configured using the user-string option.

NAS-IP Address Support in RADIUS Packets for OV Managed Switch

Nas IP Address is the attribute field in the Radius packets which is used to identify the switch that sends the radius packets and used for the accounting process.

But in OV managed switches, the interface through which the radius server will be reached is the "VPN IP Address". In order to avoid this case, use the command **aaa radius nas-ip-address** command as follows.

The following is priority when **aaa radius nas-ip-address** is configured as "default". For example:

```
-> aaa radius nas-ip-address default
```

- If IP Managed Interface is configured for Radius.
- If Loopback0 is configured.
- IP from IP stack (through which the radius server is reachable).

The following is priority when **aaa radius nas-ip-address** is configured for a local IP "12.12.12.12". For example:

```
-> aaa radius nas-ip-address local-ip 12.12.12.12
```

- If IP Managed Interface is configured for Radius.
- If Loopback0 is configured.
- User configured ip-address in CLI command.

The following is priority when **aaa radius nas-ip-address** is configured for a local IP . For example:

```
-> aaa radius nas-ip-address local-ip
```

- If IP Managed Interface is configured for Radius.
- If Loopback0 is configured.
- If DHCP Client Interface is configured.
- IP from IP stack (through which the radius server is reachable).

To display the global AAA attribute values, use the **show aaa radius config** command. For example.,

```
-> show aaa radius config
RADIUS client attributes:
  NAS identifier = default
  NAS IP Address = default
```

Configuring Case Sensitive MAC Address Authentication for RADIUS

Case sensitive MAC address Authentication supports RADIUS server authentication for supplicant, non-supplicant devices, and captive portal users.

The MAC address is sent as a part of RADIUS packets. The following data is sent as lowercase when MAC address format is selected as lowercase using the **mac-address-format lowercase** keywords:

- user-name in Access-Request and Accounting-Request
- password in Access-Request
- Accounting-Session-ID in Accounting-Request
- Calling-Station-ID in Access-Request packet.

The **aaa radius-server** command configures or modifies RADIUS server attributes with different options for Authenticated Switch Access or 802.1X port access control.

Case-sensitive MAC address authentication can be enabled using the **mac-address-format-status** option along with **aaa radius-server** command as follows:

```
-> aaa radius-server "Server1" mac-address-format-status enable
```

To specify that the MAC address format and other IDs sent to RADIUS server will be in uppercase, use the command as follows:

```
-> aaa radius-server "Server1" mac-address-format-status enable  
mac-address-format uppercase
```

The **mac-address-format** can be applied only when **mac-address-format-status** is enabled.

Configuring NAS Port for RADIUS Authentication and Accounting

A Network Access Server (NAS) operates as a client of RADIUS. The client is responsible for passing user information along with various RADIUS attributes to the designated RADIUS server, and then act on the response returned.

AOS currently supports configuration of NAS port, NAS port ID, and NAS port type to contain in the access request and accounting request packet sent to the RADIUS server.

This feature is supported for 802.1x supplicant or non-supplicant clients, and Authenticated Switch Access users, that is, management sessions like FTP, telnet, HTTP, console, HTTPS, and SSH.

Note.

- Authentication and accounting server must be configured as RADIUS for 802.1x supplicant clients, non-supplicant clients, and ASA users prior to NAS port configuration. For more information on configuring authentication and accounting server, see chapter “AAA Commands” in the *OmniSwitch AOS Release 6 CLI Reference Guide*.
- This feature is not supported when the authentication server and accounting server is configured as LDAP server, TACACS+ server, or ACE server.

To configure NAS port configuration for the RADIUS, use the **aaa radius-server** command. For example, the following command configures NAS port ID as ‘enable’ with NAS port type ‘async’:

```
-> aaa radius-server pubs2 host 1.1.1.1 key 762fefc9f0a32227 nas-port-id enable
nas-port-type async
```

All the users that get authenticated through the above RADIUS server uses the specified NAS port attributes for authentication and accounting.

Note. NAS port and NAS port ID configurations are mutually exclusive. Either NAS port or NAS port ID can be configured at a time for the RADIUS server.

Use **show aaa server** and **show configuration snapshot aaa** commands to view the NAS port, NAS port ID, and NAS port type attributes configured for the RADIUS server. However, NAS port configuration is not displayed when these attributes are configured with default value.

The following table shows the behavior of the NAS port configuration for supplicant or non-supplicant clients, and ASA users.

| NAS Port Configuration | For Supplicant or Non-supplicant clients | For ASA Users |
|-------------------------------|---|--|
| When NAS port is 'default' | Access request/accounting request packet is sent with NAS port value as 77. NAS port ID will be 'disable' and will not be sent in the access request/accounting request packet. | Access request/accounting request packet is sent with NAS port value as 77. NAS port ID will be 'disable' and will not be sent in the access request/accounting request packet. |
| When NAS port is 'ifindex' | Authenticating port is converted to ifIndex (slot*1000+port) and sent using the NAS port attribute. NAS port ID will be 'disable'. | NAS port attribute is sent as 0. NAS port ID will be 'disable'. |
| When NAS port ID is 'enable' | Authenticating port is converted to ifIndex (slot*1000+port) and sent using NAS port ID attribute. NAS port attribute will not be applicable and will not be sent in the access request/accounting request packet. | NAS port ID is sent as 0. NAS port attribute will not be applicable and will not be sent in the access request/accounting request packet. |
| When NAS port ID is 'disable' | Access request/accounting request packet is sent with NAS port value as 77. NAS port ID will be 'disable' and will not be sent in the access request/accounting request packet. | Access request/accounting request packet is sent with NAS port value as 77. NAS port ID will be 'disable' and will not be sent in the access request/accounting request packet. |

Configuring Unique Session ID for RADIUS Accounting

RADIUS Accounting Session ID feature maintains a unique session ID in RADIUS accounting for 802.1x supplicant or non-supplicant clients, captive portal users, and management sessions like FTP, telnet, HTTP, console, HTTPS, and SSH.

When accounting is configured to use RADIUS accounting,

- At the start of service delivery, an accounting Start packet is generated describing the type of service being delivered and the client it is being delivered to, and the information is then sent to the RADIUS Accounting server.
- The RADIUS server sends an acknowledgment that the packet has been received.
- At the end of service delivery, the client generates an accounting Stop packet describing the type of service that was delivered. This information is sent to the RADIUS accounting server, which sends back an acknowledgment that the packet has been received.

When unique session ID for RADIUS accounting is enabled, RADIUS attributes carry the specific authentication, authorization, and accounting details for the request and response along with the Acct-Session-ID, which gives an unique accounting ID. The unique accounting ID helps to match the start and stop records in a log file. The start, stop, and interim records for a given session must have the same Acct-Session-ID.

Note.

- Authentication server and accounting server must be configured as RADIUS prior to unique session ID configuration. For more information on configuring authentication and accounting server for 802.1x supplicant clients, 802.1x non-supplicant clients, captive portal users, and ASA users, see chapter “AAA Commands” in the *OmniSwitch AOS Release 6 CLI Reference Guide*.
- This feature is not supported when the authentication and accounting server is configured as LDAP server, TACACS+ server, or ACE server.
- This feature is not supported for local accounting.

When accounting is enabled, a unique accounting session ID is generated.

For supplicant or non-supplicant client, and captive portal users:

- When accounting is enabled, accounting session ID is generated with the combination of MAC address and time stamp.
- When accounting is disabled, MAC address is generated as the accounting session ID.

Note. When non-supplicant gets passed without authentication, the accounting session ID is generated with the combination of MAC address and 0.

For example,

```
-> 802.1x 2/5 non-supplicant policy pass vlan 10 block
```

For management sessions (FTP, telnet, HTTP, console, HTTPS, and SSH):

- When accounting is enabled, accounting session ID is generated with the combination of virtual MAC address and time stamp (time stamp is based on the user name and the session number), and passed to the RADIUS server.
- When accounting is disabled, virtual MAC address is generated as the accounting session ID.

To enable session ID for RADIUS accounting, use the **aaa radius-server** command as shown. By default, accounting session ID is disabled.

```
-> aaa radius-server pubs2 host 1.1.1.1 key 762fefc9f0a32227 unique-acct
session-id enable
```

When a new session is established, a "START" packet is sent and a session ID is generated. For example, if MAC address is 00:00:00:00:00:01, then session ID will be 000000000001-132434 where 132434 is time stamp for supplicant or non-supplicant client. For management sessions, the session ID will be in Virtual_mac_address-TimeStamp format.

To disable session ID for RADIUS accounting, use the disable option as shown:

```
-> aaa radius-server pubs2 host 1.1.1.1 key 762fefc9f0a32227 unique-acct-
session-id disable
```

When a user logs out or a session is disabled, an exit event is triggered and a "STOP" packet is sent. For example, if MAC address is 00:00:00:00:00:01, then session ID will be 000000000001. For management sessions, the session ID will be in Virtual_mac_address format.

Use **show configuration snapshot aaa** and **show aaa server** commands to view the unique session ID configuration.

Acct-Input-Gigawords and Acct-Output-Gigawords in RADIUS Accounting Packets

Acct-Input-Octets (type-42) and Acct-Output-Octets (type-43) are sent to the RADIUS Server in accounting packets. These statistics are used by the service providers for billing of users.

Acct-Input-Octets and Acct-Output-Octets fields support a maximum value of 4GB ($2^{32}-1=4294967295$). Whenever a user uses more than 4GB, the exact count of usage is lost.

Acct-Input-Gigawords (type-52) and Acct-Output-Gigawords (type-53) attributes indicate how many times the Acct-Input-Octets and Acct-Output-Octets counter has wrapped the 4GB traffic over the course the service being provided.

Whenever the input octets and output octets exceed $2^{32}-1$ bytes, before sending accounting packet to the RADIUS Server, these octets are converted into multiples of 4GB and are sent in Acct-Input-Gigawords (type-52) and Acct-Output-Gigawords (type-53) attributes. For every 4GB traffic, the value is incremented and the remaining traffic is displayed in Acct-Input-Octets and Acct-Output-Octets attribute.

For example,

A) If input octets = 5368711570

Acct-Input-Gigawords = $5368711570 / (2^{32}-1) = 1$ (4GB)

Acct-Input-Octets = $5368711570 \% (2^{32}-1) = 1073744274$

B) If output Octets = 13958643712

Acct-Output-Gigawords = $13958643712 / (2^{32}-1) = 3$ (12GB)

Acct-Output-Octets = $13958643712 \% (2^{32}-1) = 1073741824$

Note. Acct-Input-Gigawords and Acct-Output-Gigawords are sent in Interim-Update and Periodic-Interim-Update, and Logout Messages.

Acct-Input-Gigawords and Acct-Output-Gigawords are sent in accounting packets for supplicant and non-supplicant users only.

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 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**, **hash-key**, or **prompt-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**. The shared secret must be configured the same as on the server.

```
-> aaa radius-server rad1 host 10.10.2.1 10.10.3.5 key amadeus
```

Use the **hash-key** option to enter the secret key in an encrypted format. The maximum length of the hash-key is 128 characters. If 'key' and 'hash-key' parameters are configured at a time, hash-key value is given priority over key.

For example,

```
-> aaa radius-server rad1 host 10.10.2.1 hash-key testabc
```

An option **prompt-key** is provided which can be used to enter the secret key in a masked format rather than as clear text. When this option is used, a prompt appears prompting to enter the secret key. Secret key needs to be re-entered, and only if both the entries match, command is accepted. Key provided in this mode is not displayed on the CLI as text.

For example,

```
-> aaa radius-server rad1 prompt-key host 10.145.59.78
Enter key:  *****
Re-enter key: *****
```

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 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 36-3](#).

To remove a RADIUS server, use the **no** form of the command.

```
-> no aaa radius-server rad1
```

You can delete only one server at a time.

Configuring RADIUS Server Polling

Note. This command is deprecated and replaced with [aaa radius-health-check](#) command. Refer section [“Configuring RADIUS Health Check”](#) on page 36-26.

This feature provides the functionality to test the reachability of RADIUS server from the AOS Switch by enabling server polling, which polls all the configured RADIUS servers periodically to obtain the server status (Up/Down).

To enable polling of the RADIUS server, use [802.1x server-polling enable](#) command. By default, RADIUS server polling is disabled.

```
-> 802.1x server-polling enable
```

To disable polling of the RADIUS server, use [802.1x server-polling disable](#) command.

```
-> 802.1x server-polling disable
```

show aaa server command displays the reachability status of different RADIUS servers configured on the switch.

Configuring RADIUS Health Check

This feature allows to poll individual RADIUS servers for their reachability. The RADIUS Health Check functionality allows to:

- Configure the RADIUS server polling on a per RADIUS server basis.
- Configure the polling interval on a per RADIUS server basis.
- Configure the user name and password for the RADIUS polling on a per RADIUS server basis.
- Configure the action to be taken when RADIUS Health check discovers the auth-server status has changed from DOWN to UP.

Note. You need to configure “port bounce”, to bounce the port for non-suppliant re-authentication, when auth-server comes up.

To configure the RADIUS Health check for a RADIUS server, use [aaa radius-health-check](#) command.

It is recommended that Radius Health Check is enabled on all the radius servers configured for 802.1x and MAC-authentication in the system. This improves the time in which the 802.1x users are authenticated.

For example, the following command enables the radius health check for the radius server rad1 and polls the server every 700 sec with the username “admin” and password “Password1”. Since the failover parameter is enabled, it takes action if the authentication server status is changed from DOWN to UP.

```
-> aaa radius-health-check rad1 status enable polling-interval 700 username  
admin password Password1 failover enable
```

Note. When the RADIUS Health check determines the auth-server has changed status from down to up, then only the data clients that are in auth-server down UNP list are re-authenticated when failover param-

ter is enabled. The auth-server down voice clients will remain in the auth-server down UNP list in order to not disturb the ongoing sessions.

The maximum time before which the failover mechanism gets initiated from the time the server is up will be 20 seconds more than the configured polling interval. Since the polling frequency for a radius server may have lag of 20 seconds from the polling interval.

You can verify the radius health check configuration information of radius servers by using the **show aaa radius-health-check config** command.

RADIUS Server Statistics

The statistics for every RADIUS server configured on the OmniSwitch can be viewed. This allows to understand the transactions between switch and configured RADIUS server. The following statistics per RADIUS server is displayed:

- Global Information
- Authorization Statistics
- Authentication Statistics
- Accounting Statistics
- BYOD Statistics

Global Information

Global information displays the status of configured primary and backup server. The information such as the server uptime, server downtime, and number of time the server status has changed from up-down and down-up are displayed.

The global information of the configured RADIUS server can be viewed using the **show aaa server** CLI command. For example:

```
-> show aaa server

Server name = Server1
Server type      = RADIUS,
IP Address 1    = 172.21.160.29,
Retry number    = 3,
Time out (sec)  = 2,
Authentication port = 1812,
Accounting port  = 1813,
Nas port        = default,
Nas port id     = disable,
Nas port type   = ethernet,
Unique Acct Session Id = disable,
Health Check Status = DISABLED,
Server oper status = UNKNOWN,
Primary oper status = UNKNOWN,
Polling interval = 60,
User name       = admin,
Failover Status  = DISABLED
Primary server
  Server uptime      = -,
```

```
Server downtime      = MAR 23 2000 01:46:45 ,
Nb server up-down    = 0 ,
Nb server down-up    = 0 ,
Nb Rx Dropped        = 0 ,
Backup server
Server uptime        = MAR 23 2000 01:47:00 ,
Server downtime      = - ,
Nb server up-down    = 0 ,
Nb server down-up    = 0 ,
Nb Rx Dropped        = 0
```

Note. The global information (Primary and Backup server) is displayed only if RADIUS health-check is enabled. Use the [aaa radius-health-check](#) CLI command to enable RADIUS health-check on the OmniSwitch.

Authorization Statistics

Authorization is RADIUS authentication used for ASA users who connect to the switch through console, Telnet, FTP, SSH, SNMP, HTTP, and HTTPS when the configured authentication or accounting server for these type of users is set to RADIUS server.

The following authorization information is displayed for each RADIUS server:

- Number of Access-Request
- Number of Access-Response
- Number of Access-Request timed out
- Last RTT of Access-Request and Access-Response
- Minimum RTT of Access-Request and Access-Response for last week
- Average RTT of Access-Request and Access-Response for last week
- Maximum RTT of Access-Request and Access-Response for last week

Authentication Statistics

Authentication is RADIUS authentication used for network users or client who use RADIUS server for authentication.

The following authentication information is displayed for each RADIUS server:

- Number of Access-Request
- Number of Access-Response
- Number of Access-Challenge
- Number of Access-Accept
- Number of Access-Reject
- Number of Access-Request timed out
- Last RTT of Access-Request and Access-Response

- Minimum RTT of Access-Request and Access-Response for last week
- Average RTT of Access-Request and Access-Response for last week
- Maximum RTT of Access-Request and Access-Response for last week

Accounting Statistics

Authentication is RADIUS authentication used for global accounting of configured RADIUS servers.

The following information is displayed for each server:

- Number of Accounting-Request
- Number of Accounting-Response
- Number of Accounting-Request timed out
- Last RTT of Accounting-Request and Access-Response
- Minimum RTT of Accounting-Request or Access-Response for last week
- Average RTT of Accounting-Request or Access-Response for last week
- Maximum RTT of Accounting-Request or Access-Response for last week

BYOD Statistics

The following BYOD statistics for each RADIUS server is displayed:

Number of Change of Authorization (CoA) Request (COA-Req)

Number of Change of Authorization (CoA) Acknowledgment (COA-ACK)

Number of Change of Authorization (CoA) Non-Acknowledgment (COA-NACK)

Number of Disconnect Messages Request (DM-Req)

Number of Disconnect Messages Acknowledgment Request (DM-ACK)

Number of Disconnect Messages Non-Acknowledgment Request (DM-NACK)

Viewing the RADIUS Server Statistics

To view the Authorization, Authentication, Accounting, and BYOD statistics for the configured RADIUS servers, use the [show radius-server statistics](#) CLI command. For example:

```
-> show radius-server statistics

Server name = rad1
Authorization stats:
  Nb Access-Req      : 0
  Nb Access-Res      : 0
  Nb Acc-Req Time    : 0
  Last RTT           : 0
  Last week min RTT : 0
  Last week max RTT : 0
  Last week avg RTT : 0
Authentication stats:
  Nb Access-Req      : 0
```

```
Nb Access-Res      : 0
Nb Access-Chal     : 0
Nb Access-Accept   : 0
Nb Access-Reject   : 0
Nb Acc-Req Time    : 0
Last RTT           : 0
Last week min RTT  : 0
Last week max RTT  : 0
Last week avg RTT  : 0
Accounting stats:
Nb Account-Req     : 0
Nb Account-Res     : 0
Nb Accnt-Req Time  : 0
Last RTT           : 0
Last week min RTT  : 0
Last week max RTT  : 0
Last week avg RTT  : 0
BYOD stats:
Nb COA-Req         : 0
Nb COA_ACK         : 0
Nb COA_NACK        : 0
Nb DM-Req          : 0
Nb DM_ACK          : 0
Nb DM_NACK         : 0
Nb Rx Dropped      : 0
Last clear timestamp : -
```

Clearing the RADIUS Server Statistics

The RADIUS server statistics can be cleared using the [clear radius-server statistics](#) CLI command. For example:

```
-> clear radius-server statistics
```

Note. For more information on the CLI, refer the *OmniSwitch AOS Release 6 CLI Reference Guide*.

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 network 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 as 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. However, the implementation of authorization in TACACS+ server is based on the status of **aaa tacacs command-authorization** command. If the command is enabled, the authorization of every command executed on the switch is command based. CLI commands executed on the switch are sent to the TACACS+ server for authorization, along with mode of operation (read or read-write). After authorization, the server sends the response message to the TACACS+ client. If the command is disabled, then in the TACACS+ server the authorization is based on partition-management family, that is, partition-management family is sent for authorization.

When an user configures or modifies a TACACS+ server for Authenticated Switch Access using **aaa tacacs+server** command, authorization is passed on to the AAA task for authorization with a server-wait-time of only 5 seconds. However, if the TACACS server timeout is configured to a value greater than 5 seconds and multiple servers are configured, then the TACACS server takes longer time than 5 seconds to respond to authorization request. In such scenario, The CLI task is timed out on the TACACS+ server and a positive response for the previous authorization request is assigned to the next authorization request. Thus increasing the risk of authorizing server access to an unauthorized user. To avoid such unauthorized access, user can configure the server-wait-time of TACACS+ server during command authorization process using **aaa tacacs server-wait-time**.

- **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—Indicates the service is about to begin.

Stop Records—Indicates the services has terminated.

Update Records—Indicates the services are still being performed.

TACACS+ Client Limitations

The following limitations 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 is not supported. Authorization is 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 is not supported 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 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**. The shared secret must be configured the same as on the server.

```
-> aaa tacacs+-server tacl host 10.10.5.2 10.10.5.5 key otna
```

An option **prompt-key** is provided which can be used to enter the secret key in a masked format rather than as clear text. When this option is used, a prompt appears prompting to enter the secret key. Secret key needs to be re-entered, and only if both the entries match, command is accepted. Key provided in this mode is not displayed on the CLI as text.

For example,

```
-> aaa tacacs+-server tacl prompt-key host 10.135.6.219
Enter key:  *****
Re-enter key: *****
```

To modify a TACACS+ server, enter the server name and the desired parameter to be modified.

```
-> aaa tacacs+-server tacl key tnelc
```

If you are modifying the server and have 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 36-3](#).

To remove a TACACS+ server, use the **no** form of the command:

```
-> no aaa tacacs+-server tacl
```

You can delete only one server 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 can 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. Enable the schema checking 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 36-45](#) 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 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 must be part of the object class. |
| ou: <organizationalUnit name> | Defines the name of the organizational unit. |
| <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 looks 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 must be created in compliance with the LDAP protocol naming conventions.

Distinguished names are constructed from Relative Distinguished Names (RDNs). These related entries share only one-attribute value with a DN. RDNs are the components of DNs, and DNs are string representations of entry names in the directory servers.

Distinguished names consist of descriptive information about the entries. Generally, DNs include full names of individuals in a network, their E-mail addresses, TCP/IP addresses, and related attributes such as department name to distinguish the DNs. 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 the values that are allowed for a particular attribute. For example, (c=US), where country is the attribute and US is the value. Extra consideration for attribute language codes is necessary if entries are made in more than one language.

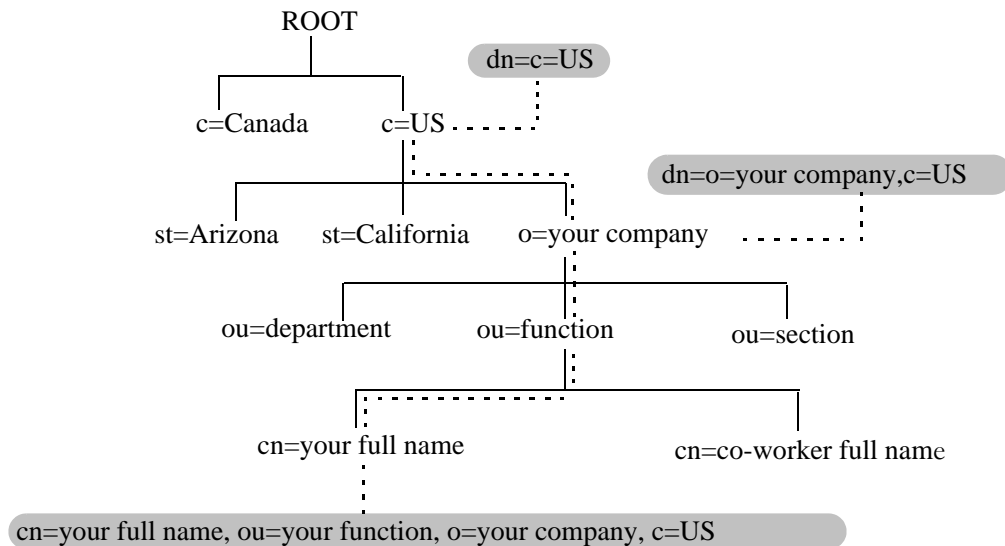
Entries are based on the physical locations and established policies in a Directory Information Tree (DIT); the DN locates an entry in the hierarchy of the tree. Alias entries pointing to other entries can also be used to circumvent the hierarchy during searches for entries.

Once a directory is set up, DN attributes must 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, 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 Information 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 must 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 retrieves the 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 value.

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 compares the 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 also sorts the 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, that is, **ldaps://**.)

LDAP URLs are entered in the command line of any web browser 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 in the previous sections.

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>
```

Example:

```
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). |
| <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 can include one or more of the following operational parameters:

- Login Restrictions
- Change Password
- Check Password Syntax
- Password Minimum Length
- Send Expiration Warnings
- Password History
- Account Lockout
- Reset Password Failure Count
- LDAP Error Messages (for example, invalid User name/Password, Server Data Error, and so on.)

For instructions on installing LDAP-enabled directory servers, refer to the vendor-specific instructions.

Directory Server Schema for LDAP Authentication

Modify the object classes and attributes accordingly to include LDAP authentication in the network (object classes and attributes are used 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. Configure server schema extensions 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 can 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 in to 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 AOS Release 6 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 logins. 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 user name 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.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 logouts. The same fields as above (separated by carriage returns “\n”) are contained in the Logout log. A different carriage return such as the # sign can 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 user name client entered to log in: variable length digits.

Fields For Layer 2 Authentication Only

- Number of bytes received on the port during the client session from login to logout: variable length digits.
- Number of bytes sent on the port during the client session from login to logout: variable length digits.
- Number of frames received on the port during the client session from login to logout: variable length digits.
- Number of frames sent on the port during the client session from login to logout: variable length digits.

AccountFailTime

The AccountFailTime attribute log records the time stamp and accounting information of unsuccessful user logins. 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 can be used in some situations. Additionally, these fields are included but apply only to the Login Fail log.

- User account ID or user name 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 can 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** are 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 are used for recording history records (not logging). For example:

```
-> aaa accounting session rad1 ldap2
```

In this example, both the **rad1** and **ldap2** servers are used for history only. Dynamic logging does 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 connection. The related attribute in the server is bop-logged users.

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 on to the switch identified by bop-basemac. |

Each switch that is connected to the LDAP-enabled directory server has 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 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, you can configure a backup host name or IP address, as well as the number of retransmit tries, the timeout for authentication requests, and whether a secure Socket Layer (SSL) is enabled between the switch and the server.

Note. Configure the server 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.

An option **prompt-password** is provided which can be used to enter the super-user password in a masked format rather than as clear text. When this option is used, a password prompt appears prompting to enter the super-user password. Password needs to be re-entered, and only if both the passwords match, command is accepted. Password provided in this mode is not displayed on the CLI as text.

For example,

```
-> aaa ldap-server topanga5 host 10.10.3.4 dn cn=manager prompt-password base c=us
retransmit 4
Enter Password:  *****
Re-enter Password:  *****
```

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 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

You can set up a Secure Socket Layer (SSL) on the server for additional security. When SSL is enabled, the identity of the server is 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 certificate must be transferred to the switch through FTP to the /flash/certified or /flash/working directory and must 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 36-5](#). For more information about the output of this command, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

37 Configuring 802.1X

Physical devices attached to a LAN port on the switch through a point-to-point LAN connection is 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 AOS Release 6 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 37-9](#)
- [“Enabling 802.1X on Ports” on page 37-9](#)
- [“Setting 802.1X Switch Parameters” on page 37-9](#)
- [“Configuring 802.1X Port Parameters” on page 37-10](#)
- [“Verifying the 802.1X Port Configuration” on page 37-15](#)

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 6350, 6450 |
| Maximum number of 802.1x users per NI module. | In OS6450, 256 (supplicants or non-supplicants) In OS6350, 96 (supplicants or 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 does not accept an 802.1X authentication attempt. | quiet-period | 60 sec |
| The time before an EAP Request Identity will be re-transmitted. | tx-period | 30 sec |
| Number of sec before the switch will time out an 802.1X user who is attempting to authenticate. | supp-timeout | 30 sec |
| 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 retransmits an authentication request before it times out. | max-req | 2 |
| Amount of time that must expire before a reauthentication attempt is made. | re-authperiod | 3600 sec |
| 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 37-3](#) 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 trust-radius and session-timeout parameters for the 802.1x port to enable or disable the session timeout and set the session timeout interval for MAC authenticated users. Use the **802.1x** command as follows:

```
-> 802.1x 1/1 non-supPLICANT session-timeout enable interval 14000 trust-radius enable
```

7 (Optional) Configure trust-radius and session-timeout parameters for the 802.1x port to enable or disable the session timeout and set the session timeout interval for supplicant users. Use the **802.1x** command as follows:

```
-> 802.1x 1/1 trust-radius enable
```

8 (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 (sec)        = 60,
tx-period (sec)           = 30,
```

```

supp-timeout (sec)           = 30,
server-timeout (sec)        = 30,
max-req                      = 2,
re-authperiod (sec)         = 3600,
reauthentication             = no
Supplicant polling retry count = 2
Trust-Radius                 = enabled,
isPortAP                    = yes,
Supplicant polling retry count = 2,
Captive Portal Session Limit (hrs) = 12,
Captive Portal Login Retry Count = 3,
Supplicant Bypass           = disable,
Supplicant Bypass allow-eap Branch = none,
Non-Supp reauthentication    = disabled,
Non-Supp re-authperiod (seconds) = 43200,
Non-Supp Trust-Radius        = disabled,
Captive Portal Inactivity Logout = Disabled,

```

Optional. To display the number of 802.1x users on the switch, use the [show 802.1x users](#) command:

```
-> show 802.1x users
```

| Slot Port | MAC Address | Port State | Classification Policy | Auth Failure Reason | Auth Retry Count | Last Successful Auth Time | User Name |
|--------------|-------------------|---------------|--------------------------|------------------------|---------------------|------------------------------|--------------|
| 04/05 | 00:13:72:ae:f3:1c | Connecting | | AUTHENTICATION FAILURE | 1 | - | user |

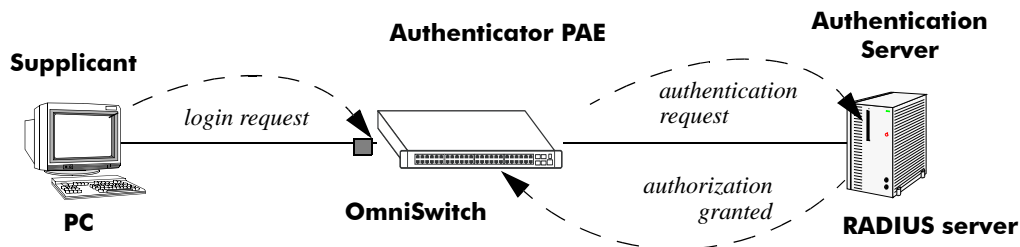
See the *OmniSwitch AOS Release 6 CLI Reference Guide* for information about the fields in this display.

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 through 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 through 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, and so on) 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 identity of the supplicant. 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.

The 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 in the previous conditions. 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 37-9](#) 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 authenticate on the port first) or an administrator may force the port to never accept any frames on the port. See [“Configuring the Port Authorization” on page 37-10](#).

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 are 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 37-12](#)).

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 alerts 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 37-12](#).

Enabling 802.1x pass-through

In order to transparently forward the 802.1x EAP frames in the switch, the switch should be set to pass-through mode. To enable the pass-through mode in the switch use the [802.1x pass-through](#) command.

Pass through mode should be enabled on the layer2 switch to allow EAP packets to be trapped on the layer3 switch for authentication.

802.1X Accounting

If servers are set up for 802.1X accounting the 802.1X authentication sessions may be logged. 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 37-12](#).

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 is used for authenticating 802.1X ports. If **rad1** becomes unavailable, the switch uses **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. 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 is set up with defaults listed in [“802.1X Defaults” on page 37-3.](#)

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 retransmits 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 is re-transmitted.
- Supplicant (or user) timeout—The time before the switch times out an 802.1X user who is attempting to authenticate. During the authentication attempt, the switch sends requests for authentication information (identity requests, challenge response, and so on.) to the supplicant (see [“Configuring the Maximum Number of Requests”](#) on page 37-11). If the supplicant does not reply to these requests, the supplicant is timed out when the timeout expires.
- Session timeout—The session timeout interval for MAC authenticated users.

The 802.1x non-supplicant session timeout is disabled by default and when enabled the default session timeout interval is set to 43200 seconds.

```
-> 802.1x 1/1 non-supplicant session-timeout enable trust-radius enable
```

When the trust-radius option is enabled, the timeout value returned in session-timeout attribute of Access-Accept message takes precedence over the configured session-timeout. The change in session timeout

interval takes effect immediately for all users that are authenticated after the configuration. For users who are already authenticated the session timeout interval takes effect only after the user is flushed out or when the user is re-authenticated again.

The 802.1x inactivity-logout for non-supPLICANT users is enabled by default. If the inactivity-logout is disabled then MAC entry would be re-programmed in the switch after MAC aging without any re-authentication initiated by the switch. This avoids the MAC address timeout for non-supPLICANT users.

```
-> 802.1x 1/1 non-supPLICANT inactivity-logout disable
```

The 802.1x supPLICANT session timeout is disabled by default and when enabled the session-timeout attribute value returned by the server is used as re-authentication time period for re-authentication.

```
-> 802.1x 1/1 trust-radius enable
```

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 sec; the transmit timeout to 25 sec; and the user timeout to 25 sec.

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 sends 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 expires. The switch then uses 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 is 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-supPLICANT, and any non-supPLICANT 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-supPLICANTS, 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-supplicants. As a result, non-supplicant policies that use MAC-based authentication are now applicable to supplicant devices and not only non-supplicant devices.

Re-authenticating an 802.1X Port

An automatic re-authentication 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 takes place periodically 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 takes place on the port every 25 sec.

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 has to 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** is 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.”](#)

Configuring 802.1x Delay Learning

To avoid 802.1X clients from getting into the auth-server-down state by attempting an early authentication before the switch is rebooted, the delay learning interval can be set for 802.1X clients. This delays the 802.1X authentication process until the switch is rebooted. By default, the delay-learning interval is set to 120 seconds. To configure the delay-learning interval, use the [802.1x delay-learning](#) command. For example:

```
-> 802.1x delay-learning 300
```

This delays the 802.1X authentication process by 300 secs during the switch reboot.

Use the [show 802.1x auth-server-down](#) command to view the configured delay learning interval.

Re-authentication Process Based on the RADIUS returned attributed Session-Timeout

The following table displays the reauthentication process based on the session-timeout configuration.

For non-supPLICANT user:

| Session-Timeout | Trus-Radius | Server Returns session-timeout attribute | Reauthentication Process |
|-----------------|-------------|--|--|
| Enable | Enable | Yes | Reauthentication happens based on server returned timer value. |
| Enable | Enable | No | Reauthentication happens based on configured timer interval. |
| Enable | Disable | Yes | Reauthentication happens based on configured timer interval. |
| Enable | Disable | No | Reauthentication happens based on configured timer interval. |
| Disable | Enable | Yes/No | No Reauthentication happens. |
| Disable | Disable | Yes/No | No Reauthentication happens. |

For supplicant user:

| Session-Timeout | Trus-Radius | Server Returns session-timeout attribute | Reauthentication Process |
|-----------------|-------------|--|--|
| Enable | Enable | Yes | Reauthentication happens based on server returned timer value. |
| Enable | Enable | No | Reauthentication happens based on configured timer interval (re-authperiod). |

| Session-Timeout | Trus-Radius | Server Returns session-timeout attribute | Reauthentication Process |
|-----------------|-------------|--|--|
| Enable | Disable | Yes | Reauthentication happens based on configured timer interval (re-authperiod). |
| Enable | Disable | No | Reauthentication happens based on configured timer interval (re-authperiod). |
| Disable | Enable | Yes/No | No Reauthentication happens. |
| Disable | Disable | Yes/No | No Reauthentication happens. |

To view the time period in which reauthentication happens, use the [show 802.1x non-suppliant detail](#) and [show 802.1x users detail](#) CLI command.

```
-> show 802.1x non-suppliant detail
```

```
Slot 1 Port 36 - has no non-suppliant to show.
```

```
Slot/Port          = 02/13
MAC Address        = 00:00:c3:de:79:b8
MAC Authen Status  = Authenticated
Classification Policy = Basic-VLAN ID
VLAN Learned       = 100
Dynamic UNP        = Disabled
Username           = User1
ReAuthPeriod       = 40
HIC Status         = Not Started
```

```
-> show 802.1x users detail
```

```
Slot/Port          = 02/13
MAC Address        = 00:00:c3:de:79:b8
Port State         = Authenticated
Classification Policy = Basic-VLAN ID
VLAN Learned       = 100
Username           = User2
ReAuthPeriod       = 60
Dynamic UNP        = Disabled
HIC Status         = Not Started
```


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 AOS Release 6 CLI Reference Guide*.

38 Managing Policy Servers

Quality of Service (QoS) policies that are configured through Alcatel'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 AOS Release 6 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 38-3](#)
- [“Modifying Policy Servers” on page 38-4](#)
- [“Verifying the Policy Server Configuration” on page 38-7](#)

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 6350, 6450 |
| 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 |

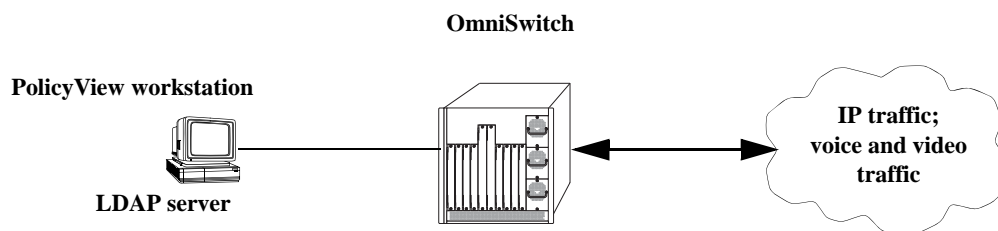
Policy Server Overview

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 39, “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 38-2](#).

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 user name 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 38-5](#).

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,
  Status               : Down,
  Preference           : 99,
  Authentication       : password,
  SSL                  : Disabled,
  login DN             : cn=DirMgr
  searchbase           : o=company
  Last load time       : 02/14/02 16:38:18
```

Note. When an OmniSwitch is having more than two policy servers configured, the highest precedence server will be identified and the configuration of the highest precedence server will be loaded to avoid the policy recache.

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 39, “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. |

39 Configuring QoS

Alcatel QoS software provides a way to manipulate data flows coming through the switch based on user-configured policies. Manipulation of the data flow (referred to as *Quality of Service* or *QoS*) can be as simple as allowing or denying traffic. It can also be as complicated as remapping 802.1p bits from a Layer 2 network to ToS values in a Layer 3 network.

While policies are 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 40, “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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Setting up global QoS parameters (see [page 39-15](#))
- Configuring QoS Ports and Queueing Schemes [page 39-24](#)
- Setting up policy components, such as policy conditions and actions (see [page 39-34](#))
- Configuring specific types of policies (see [page 39-67](#))

Note. Policies can also be configured through the PolicyView NMS application and stored on an attached LDAP server. LDAP policies are downloaded to the switch and managed through the Policy Manager feature in the switch. For more information about managing LDAP policies, see [Chapter 38, “Managing Policy Servers.”](#)

QoS Specifications

The QoS functionality described in this chapter is supported on OmniSwitch 6350, 6450 switches. Any maximum limits provided in the Specifications table are subject to available system resources.

| | |
|--|--|
| Maximum number of policy rules | 1024 (ingress and egress rules combined) |
| Maximum number of egress policy rules | 512 |
| Maximum number of policy conditions | 2048 |
| Maximum number of policy actions | 2048 |
| Maximum number of policy validity periods | 128 |
| Number of predefined QoS profiles | 16 |
| Maximum number of user-defined QoS profiles | 112 |
| Maximum number of policy services | 256 |
| Maximum number of TCP and UDP port ranges | 4 |
| Maximum number of groups | 1024 |
| Maximum number of group entries | 1024 per group (512 per service group) |
| Maximum number of port groups per policy | 8 |
| Maximum number of bandwidth shaping rules per slot | 640 |
| Maximum configurable shaper value on 10 Gig | 8 Gbps |
| 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 |
| Default value of shared buffers | 1500 |
| Shared buffers range | 0-4095 |
| CLI Command Prefix Recognition | Some QoS commands support prefix recognition. See the “Using the CLI” chapter in the <i>OmniSwitch AOS Release 6 Switch Management Guide</i> for more information. |

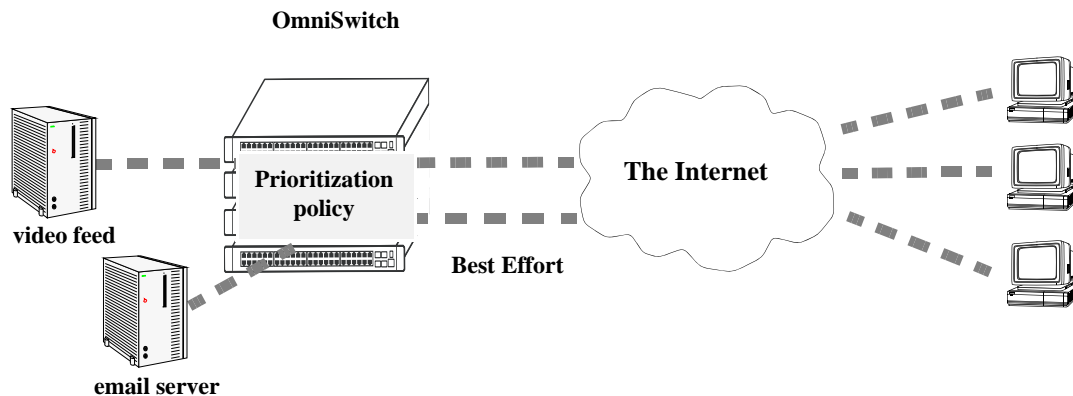
QoS General Overview

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 the mechanisms 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 can still occur. Also, adding enough bandwidth to compensate for peak load periods means at times some bandwidth is unused. In addition, adding bandwidth does not guarantee any 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 can be given to individual flows as required. Voice over IP (VoIP) traffic or mission-critical data can 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 entire link 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 can 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 examines in incoming flows, such as destination address or Type of Service (ToS) bits. The action specifies what the switch does with a flow that matches the condition; for example, it can queue the flow with a higher priority, or reset the ToS bits.

Policies can be created using one of the following methods:

- directly on the switch through the CLI or WebView
- on an external LDAP server through the PolicyView application.

The switch distinguishes between policies created on the switch and policies created on an LDAP server.

Note. Policies must be modified using the same source used to create them. Policies configured through PolicyView can be edited only through PolicyView. Policies created directly on the switch through the CLI or WebView can be edited only on the switch. However, to override policies created in PolicyView, use CLI or WebView.

This chapter discusses policy configuration using the CLI. For information about using WebView to configure the switch, see the *OmniSwitch AOS Release 6 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 QoS 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 39-41](#).
- ***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 39-24](#).

Valid Policies

The switch does not allow you to create invalid condition or action combinations; if you enter an invalid combination, an error message is displayed.

A list of valid condition and condition or action combinations is given in [“Condition Combinations” on page 39-7](#) and [“Action Combinations” on page 39-9](#).

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 39-7](#) and [“Action Combinations” on page 39-9](#)).

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.
- **User Network Profile (UNP)**—This type of policy list is associated with an Access Guardian UNP. The rules in this list are applied to ingress traffic that is classified into the user profile. See [Chapter 35, “Configuring Access Guardian,”](#) for more information.
- **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 39-7](#) and [“Action Combinations” on page 39-9](#) to determine which conditions and actions are supported.

The policy list is configured as UNP or egress list when the list is created. For more information, see [“Creating Policy Lists” on page 39-42](#).

Interaction With Other Features

QoS policies can be used to configure other switch features, such as Link Aggregation. In addition, QoS settings can affect other features in the switch; or QoS settings can require other switch features be configured in a particular way.

A summary of related features is given here:

- **Dynamic Link Aggregates**—Policies can be used to prioritize dynamic link aggregation groups. For details, see [Chapter 26, “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 24, “Configuring 802.1Q.”](#)
- **Mobile Ports**—Mobile ports are always trusted, regardless of QoS settings. For information about setting up mobile ports, see [Chapter 7, “Assigning Ports to VLANs.”](#)
- **LDAP Policy Management**—Policies can also be configured through the PolicyView application and stored on an attached LDAP server. LDAP policies can only be modified through PolicyView. For information about setting up a policy server and managing LDAP policies, see [Chapter 38, “Managing Policy Servers.”](#)

Ethernet Service (VLAN Stacking)

- **VLAN Stacking ports are always trusted and default classification is set to 802.1p. VLAN Stacking Ethernet Service**—VLAN Stacking ports are always trusted and default classification is set to 802.1p. QoS policy conditions to match the inner VLAN tag and inner 802.1p tag are available for classifying customer information contained in VLAN Stacking frames. For information about VLAN Stacking see [Chapter 9, “Configuring VLAN Stacking.”](#)
- **User Network Profiles**—The Access Guardian User Network Profile (UNP) feature provides the ability to assign a list of QoS policy rules to a profile. The rules contained in the list are applied to any device that is assigned to the UNP. For more information about policy lists, see “Policy Lists” on page 29-5 and [Chapter 35, “Configuring Access Guardian.”](#)
- **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 10, “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 can configure **source ip** and a **destination ip** for the same condition.

The following conditions are supported and can be combined with other conditions and/or actions. 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 39-42](#) 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 inner 802.1p ethertype source VLAN source VLAN group destination VLAN (multicast rules only) destination VLAN group (multicast rules only) inner source VLAN inner source VLAN group | source MAC source MAC group destination MAC destination MAC group 802.1p inner 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:

- Destination port / destination port group cannot be used in default policy list.
- 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 can 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 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.
- 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.
- To apply a Layer 2 rule to IPv6 traffic using the source or destination MAC address, 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 39-9 and [“Condition and Action Combinations”](#) on page 39-11.

For specific information about how to configure policy conditions and actions to create a policy rule, see [“Creating Policies”](#) on page 39-34.

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 can be combined with an action specifying priority.

The following actions are supported and can be combined with other actions. 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 39-42](#) 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 | Yes | No |
| No Cache (disables the logging of rule entries to the hardware cache) | Yes | No |
| Port Disable | Yes | No |
| Permanent Gateway IP | 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 gives 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 | No |
| 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 39-7 and [“Condition and Action Combinations”](#) on page 39-11.

For specific information about how to configure policy conditions and actions to create a policy rule, see [“Creating Policies”](#) on page 39-34.

Condition and Action Combinations

Conditions and actions are combined in policy rules. The CLI prevents you from configuring invalid condition or 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 or action combination restrictions can be applied depending on whether the policy rule is being applied to ingress or egress traffic. See [“Condition Combinations” on page 39-7](#) and [“Action Combinations” on page 39-9](#) 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 |
| Specifies whether log messages are sent to the console | qos log console | no |
| Specifies whether log messages are available to OmniVista applications | qos forward log | no |
| Specifies whether IP anti-spoofing is enabled on UserPorts. | qos user-port filter | yes |
| Specifies 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 shut down operation). |

Policy Action Defaults

The following are defaults for the **policy action** command:

| Description | Keyword | Default |
|---|--------------------|------------------------------|
| Whether the flow matching the rule must 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=0K |
| - peak rate and burst size | pir pbs | PIR=0, PBS=0K |

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 39-37 and “Setting the Global Default Dispositions” on page 39-16.
- *The switch network group*—The switch has a default network group, called **switch**, that includes all IP addresses configured for the switch itself. The default network group can be used in policies. See “Creating Network Groups” on page 39-51 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**, and so on. 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 or disabling QoS, global configuration includes settings such as global port parameters, default disposition for flows, and various timeouts. The type of parameters you want to configure globally depends on the type of policies you configure. For example, if you want to set up policies for 802.1p or ToS/DSCP traffic, you can configure all ports as trusted ports.

Typically, it is not required to change any of the global defaults. See [“Global QoS Defaults” on page 39-12](#) for a list of the global defaults. See [“Configuring Global QoS Parameters” on page 39-15](#) for information about configuring global parameters.

2 Configuring QoS Port Parameters. This configuration includes setting up QoS parameters on a per port basis. Typically, it is not required to change any of the port defaults. See [“QoS Port Defaults” on page 39-13](#) for a list of port defaults. See [“QoS Ports and Queues” on page 39-24](#) for information about configuring port parameters.

3 Setting Up Policies. Most QoS configuration involves setting up policies. See [“Creating Policies” on page 39-34](#).

4 Applying the Configuration. Use `qos apply` command to configure policy rule and some global parameters before they are active on the switch. See [“Applying the Configuration” on page 39-64](#).

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 attempts 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 are used to classify traffic coming into the switch.

Individual policy rules can 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 (that determines whether the switch accepts, denies, or drops 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](#).

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 39-64](#).

The default disposition for routed flows is not configurable on a global basis for the switch. Policies can be configured to deny any routed traffic through the switch.

Typically, the disposition is only configured when you are using policies for Access Control Lists (ACLs).

Set the **qos default bridged disposition** to **deny** to 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 40, “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 eight 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 39-24](#).

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 AOS Release 6 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:

| MAC Address Range | Description |
|--|--------------------------------|
| 00:80:9F:00:00:00 to 00:80:9F:FF:FF:FF | Enterprise IP Phones Range |
| 78:81:02:00:00:00 to 78:81:02:FF:FF:FF | Communications IP Phones Range |
| 00:13:FA:00:00:00 to 0:13:FA:FF:FF:FF | Lifesize IP Phones Range |
| 48-7A-55-00-00-00 to 48-7A-55-FF-FF-FF | ALE 8008 IP Phone MAC Range |

In addition to prioritizing IP phone traffic, it is also possible to prioritize non-IP phone traffic automatically. To prioritize non-IP phone traffic, add up to four MAC addresses or four ranges of MAC addresses to the predefined QoS “alaPhones” MAC address group. See [“Creating MAC Groups” on page 39-54](#) 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
```

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 can 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, can query the switch for log events; or log events can be immediately available to the PolicyView application through a CLI command. Log events can also be forwarded to the console in real time.

IP Source Filtering Drop-Log (ISF) feature enables the user to see the packets getting dropped by IP Source Filter entries. When ISF (**ip helper dhcp-snooping ip-source-filter**) is enabled on a port or VLAN, it restricts all the IP traffic on that port except the DHCP traffic and the traffic from the client, whose binding entry exists on that port. With ISF drop log feature, whenever a packet is dropped by ISF drop entry in the hardware, drops are logged in QoS log, which are displayed in **show qos log** command. This will enable the user to know which port/MAC/IP was dropped.

ISF drop logging is enabled by default. Hence if the packets are getting dropped due to ISF drop rule, packets are logged. 64 packets are logged per second.

What Information Is Logged

The **debug qos** command controls the information displayed in the log. The **qos log level** command determines how specific the log messages will be. See “[Log Detail Level](#)” on page 39-19.

By default, only the most basic QoS information is logged. The types of information that are logged includes rules, Layer 2 and Layer 3 information, and so on. For a detailed explanation about the types of information that are logged, see the *OmniSwitch AOS Release 6 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 to be displayed, use the **qos log lines** command to 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 entire QoS log is cleared. To change the log lines without clearing the log, set the log lines in the **boot.cfg** file. At the next reboot, the log is set to the specified number of lines.

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 is given in the QoS log. The **qos log level** command is associated with the **qos debug** command that determines the information to 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 39-64](#).

Note. A high log level value impacts the performance of the switch.

Forwarding Log Events

NMS applications 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 39-64](#).

If event forwarding is disabled, NMS applications can still query the QoS software for events, but the events are not sent in real time.

Forwarding Log Events to the Console

QoS log messages can 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 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 39-64](#).

Use the **swlog output flash file-size** command to configure switch logging to output logging events to the console 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 AOS Release 6 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 yubal (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
```

Use the **qos log lines**, **qos log level**, and **debug qos** commands to modify the log display. The log display can also be output to the console through the **qos log console** command or sent to the policy software in the switch (that manages policies downloaded from an LDAP server) through the **qos forward log** command.

show qos log command also displays the IP Source Filtering drop entries.

```
-> show qos log

**QOS Log**
9/16/01 18:09:18 [@18:09:18] rule ISF-DROP matched
9/16/01 18:09:18 Tagged.      802.1p 0
9/16/01 18:09:18 svlan 10 VRF (null) port 1/9
9/16/01 18:09:18 MAC 00:00:1E:1D:EE:14 -> E8:E7:32:77:BB:A2
9/16/01 18:09:18 TOS 0x00 (p255) 10.10.10.10 -> 10.10.10.100
9/16/01 18:09:18 [@18:09:18] rule ISF-DROP matched
```

Clearing the QoS Log

The QoS log can get large if invalid rules are configured on the switch, or if many 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 40, “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. 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 can bridge and route traffic to the same destination.
- Bridged IP packets are prioritized based on ToS, not 802.1p.

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. A sample output is as follows:

```
-> show qos statistics
```


| Software resources | | | | | | | | | | | | |
|--------------------|---------|------|------|-----|-------|------|---------|------|------|-----|-------|------|
| Table | Applied | | | | | | Pending | | | | | |
| | CLI | LDAP | ACLM | Blt | Total | Max | CLI | LDAP | ACLM | Blt | Total | Max |
| rules | 0 | 0 | 0 | 0 | 0 | 2048 | 0 | 0 | 0 | 0 | 0 | 2048 |
| actions | 0 | 0 | 0 | 0 | 0 | 2048 | 0 | 0 | 0 | 0 | 0 | 2048 |
| conditions | 0 | 0 | 0 | 0 | 0 | 2048 | 0 | 0 | 0 | 0 | 0 | 2048 |
| services | 0 | 0 | 0 | 0 | 0 | 256 | 0 | 0 | 0 | 0 | 0 | 256 |
| service groups | 1 | 0 | 0 | 0 | 1 | 1024 | 1 | 0 | 0 | 0 | 1 | 1024 |
| network groups | 0 | 0 | 0 | 1 | 1 | 1024 | 0 | 0 | 0 | 1 | 1 | 1024 |
| port groups | 2 | 0 | 0 | 8 | 10 | 1024 | 2 | 0 | 0 | 8 | 10 | 1024 |
| mac groups | 0 | 0 | 0 | 0 | 0 | 1024 | 0 | 0 | 0 | 0 | 0 | 1024 |
| map groups | 0 | 0 | 0 | 0 | 0 | 1024 | 0 | 0 | 0 | 0 | 0 | 1024 |
| vlan groups | 0 | 0 | 0 | 0 | 0 | 1024 | 0 | 0 | 0 | 0 | 0 | 1024 |

| Hardware resources | | | TCAM | | | Ranges | | |
|--------------------|-------|------|------|------|------|--------|------|-----|
| Slot | Slice | Unit | Used | Free | Max | Used | Free | Max |
| 1 | 0 | 0 | 0 | 1024 | 1024 | 0 | 0 | 0 |

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 39-12](#).

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 AOS Release 6 CLI Reference Guide*.

QoS Ports and Queues

Queue parameters can 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 use the same values to implement the maximum bandwidth.

Prioritizing and Queue Mapping

QoS prioritizes packets by placing them in a higher priority egress queue. As previously described, 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 39-27](#) 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 port is a UNI port, considering the SAP profile information, the DHCP packet is parsed to get the 802.1p tag.
- 3 The internal priority, 802.1p and DSCP value are set based on the actions configured for this rule. The rule can be set with the following criteria:
 - a Internal priority only.
 - b Set the 802.1p and internal priority (always overwrite the priority and ToS/DSCP action).
 - c Set the ToS/DSCP and the internal priority only when priority is not specified.
 - d 802.1p/ToS/DSCP mapping (map 802.1p/tos/dscp to 802.1p/tos/dscp).
 - e Modify the packet outer 802.1p value when applicable. Enqueue the packet to the appropriate egress port queue
- 4 If the port is trusted, only the internal priority value is retrieved based on the default classification of the port and the DHCP packet 802.1p or DSCP fields.
- 5 If a port is untrusted, the internal priority and 802.1p value is set based on the port default 802.1p setting and the DSCP value is set based on the port default DSCP setting.
- 6 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 39-33](#) for more information.

7 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 39-25](#) for more information.

8 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.

9 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.

10 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 39-28](#) for more information.

11 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 | OmniSwitch 6350, 6450 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 must trust.
- 2 Define policy conditions for the port group; one condition for each L2 priority (802.1p) value.
- 3 Define policy actions that stamps 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

```

For pure Layer 2 packets, trusted ports 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 sends four times as much traffic as a queue with a weight of one.
- On OmniSwitch, 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 to apply to all switch ports (see [“Setting the Global Default Servicing Mode” on page 39-16](#)) or configure the servicing mode on an individual port basis (see [“Configuring the Servicing Mode for a Port” on page 39-27](#)).

The QoS servicing mode only applies to destination ports because it is 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 39-16](#) for more information.

Note the following when configuring the port servicing mode:

- Servicing mode configurations can be applied to a maximum of five ports per slot.
- 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, 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 39-57 and “Policy Applications” on page 39-67 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 39-28 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 64 Kbps. Any value specified that is not a multiple of 64K is rounded up to the next highest multiple of 64 K.
- 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
```

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 39-69](#) 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. When a switch receives a yellow packet with the DEI bit set and DEI mapping is enabled, the packet is mapped to an internal drop precedence or yellow color marking for the switch.

The switch can be set globally so that DEI bit marking and mapping is enabled for all ports. Individual ports can 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 can 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 AOS Release 6 CLI Reference Guide* for more information about these commands.

Equal Scheduling For Yellow Traffic

This functionality enables equal scheduling of yellow traffic by configuring priority value for yellow traffic on OmniSwitch. This configuration is global and ensures fair sharing of yellow traffic. When the egress link is congested, the incoming yellow traffic on the link on all egress queues is equally shared based on the ingress traffic.

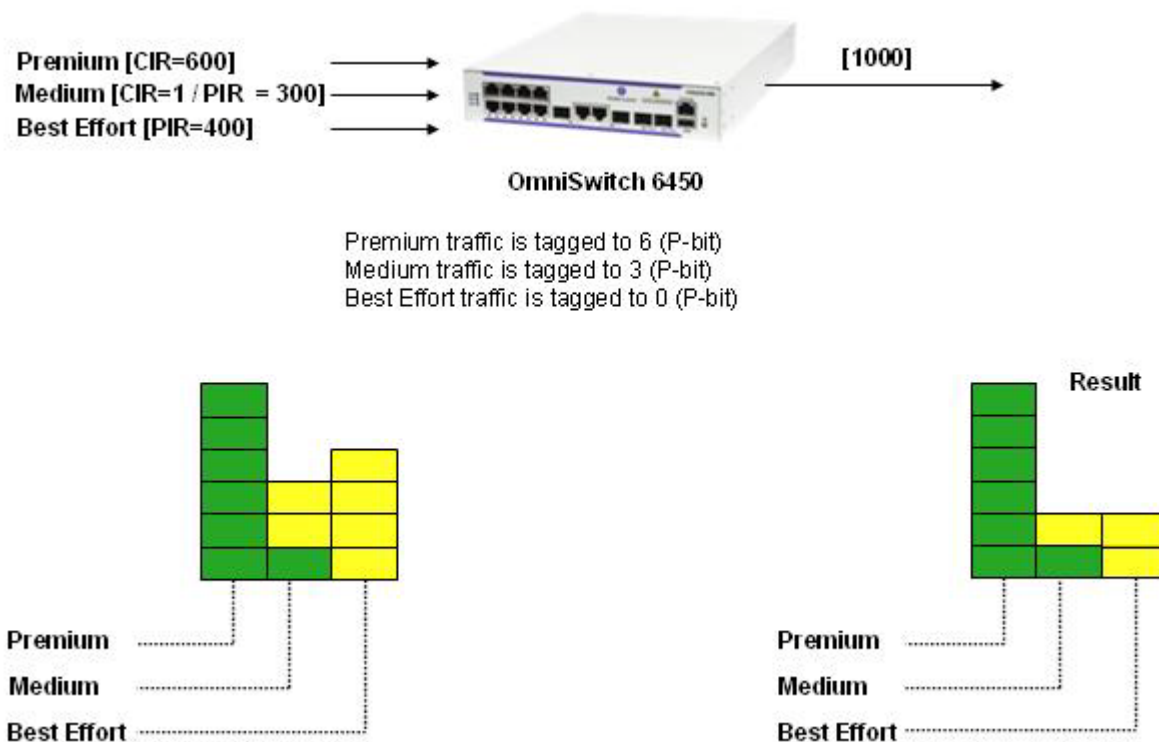
For example, consider the following scenario:

Total egress link bandwidth: 1000 Mbps

1. Premium with priority 6 and CIR=600 Mbps PIR=600 Mbps - All frames are green with priority 6.
2. Medium with priority 3 and CIR=100 Mbps PIR=300 Mbps - Frames from 100 Mbps - 300 Mbps will be yellow.
3. Best effort with priority 0 and CIR=0 PIR = 400 Mbps - All frames are yellow with priority 0.

In this scenario, all premium traffic with highest priority 6 is allowed on the egress link without any drop. The medium with priority 3 takes up the next 300 Mbps on the egress link leaving 100 Mbps on the egress link for best effort traffic with priority 0. The high priority (medium) yellow traffic is given preference over the best effort yellow traffic.

The following diagram depicts the traffic behavior when there is no equal scheduling of yellow traffic:



Traffic behavior without equal scheduling configuration

When equal scheduling of yellow traffic is configured, yellow traffic on all the queues is equally shared between medium and best effort.

When equal scheduling is configured for yellow traffic, medium with priority 3 gets 100 Mbps green traffic and 100 Mbps yellow traffic; the best effort gets 200 Mbps yellow traffic.

The equal scheduling is calculated according to the following formula:

Yellow traffic generated by Premium traffic = 0 Mbps

Yellow traffic generated by Medium traffic = 200 Mbps

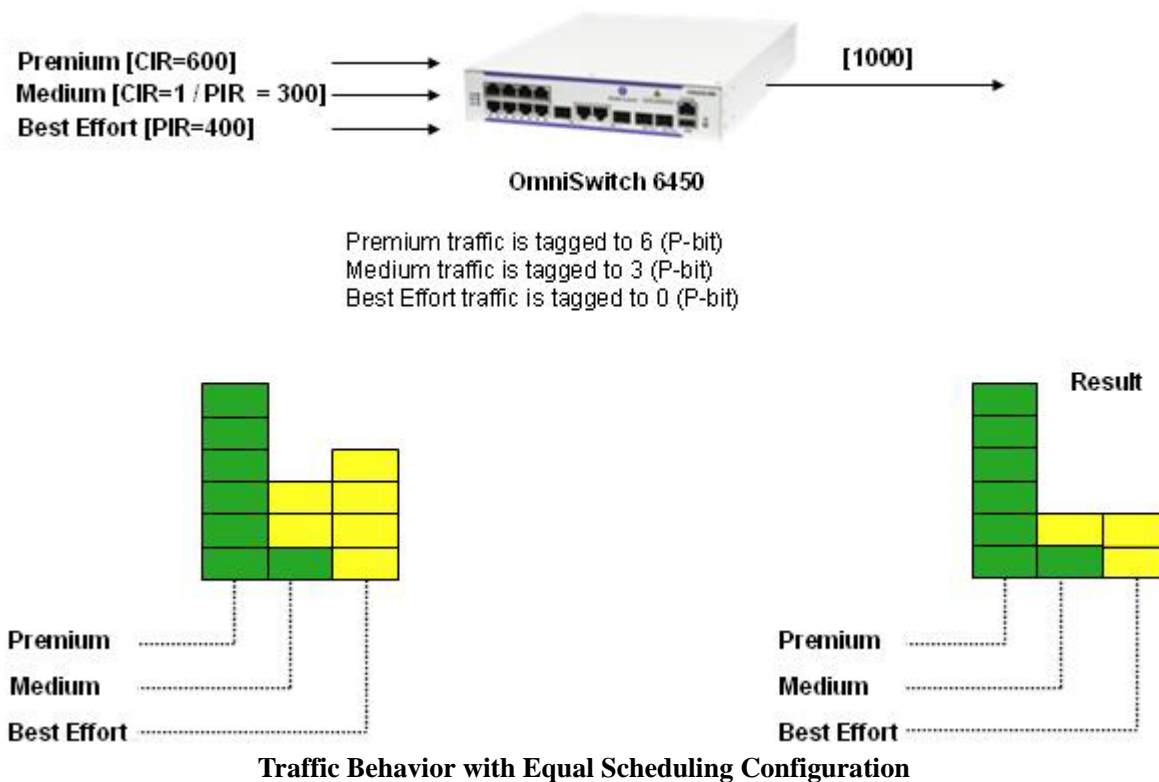
Yellow traffic generated by Best Effort traffic = 400 Mbps

The remaining bandwidth is distributed in the ratio 0:200:400 (0:1:2 between Medium and Best Effort traffic)

Medium gets $1/3 * 300$ Mbps = 100 Mbps

Best Effort gets $2/3 * 300$ Mbps = 200 Mbps

The following diagram depicts the traffic behavior after configuring equal scheduling of yellow traffic:



To Configure Equal Scheduling of Yellow Traffic

Configure equal scheduling of yellow traffic using the **QoS force-yellow-priority** command. By default, the priority value is set to 0. The following example sets the priority of the yellow traffic to 5.

```
-> qos force-yellow-priority 5
```

Use the **no** form of the command to remove the equal scheduling of yellow traffic. This removes the priority configured for the yellow traffic. For example:

```
-> qos no force-yellow-priority
```

Yellow frames are identified by the “drop precedence” field in qos-profile table. For yellow frames, drop precedence field will be set as 1.

The configured 802.1P/DSCP priority values will be remarked into yellow frames and processed on the egress queue. The following example sets the priority of 802.1P profile parameter to 5.

```
-> qos force-yellow-802.1p 5
```

This CLI command forcefully sets 802.1p value of yellow packets to 5 in global level. The default priority assigned is none. The priority value ranges from 0 (lowest) to 7 (highest).

Use the **no** form of the command to remove the 802.1P profile. This removes the priority configured for 802.1P profile globally. For example:

```
-> qos no force-yellow-802.1p
```

To configure the DSCP priority value, use the following command.

```
-> qos force-yellow-dscp 5
```

This CLI command forcefully sets DSCP priority value globally. The default priority is none. The priority value ranges from 0 (lowest) to 63 (highest).

Use the **no** form of the command to restore the DSCP priority value. For example:

```
-> qos no force-yellow-dscp
```

To view the modified profile table setting enabled for yellow frames along with the priority value, use the command [show qos config](#).

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 24, “Configuring 802.1Q.”](#)

Mobile ports are also always trusted; however, mobile ports can or do not accept Q-tagged traffic.

Note . Mobile ports are not Q-tagged in the same manner as fixed ports; however, a mobile port joins 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 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 can be set globally so that all ports are trusted. Individual ports can be configured to override the global setting.

QoS Profiles for Trusted and Untrusted Ports

To ensure consistent QoS treatment in a stack system, the QoS profiles are configured so that all the QoS profile table entries are same in all the NIs of the stack.

All the QoS profiles are globally managed in the CMM. Any changes in the QoS policies on the CMM are notified to all the NIs in stack. There are two types of QoS profiles:

- Pre-defined QoS profiles
- User defined QoS profiles

A total of 16 pre-defined QoS profiles are statically created at system initialization or bootup. Now a maximum of 112 user-defined QoS profiles can be created.

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 39-64](#) for information about the **qos apply** command.

Using Trusted Ports With Policies

It is important to know whether the port is trusted or untrusted when classifying traffic with 802.1p bits. If the policy condition specifies 802.1p, the switch must be able to recognize 802.1p bits. (The trusted port must also be 802.1Q-tagged as described in [“Setting the DEI Bit” on page 39-29](#).)

The 802.1p bits can 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 recognizes 802.1p bits. A policy condition (**Traffic**) is then created to classify traffic containing 802.1p bits set to four ingressing on port 2 on slot 3. When the traffic egresses the switch, the **policy action (SetBits)** specifies that the bits are reset to 7. 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 39-64](#).

For actions that set 802.1p bits, a limited set of policy conditions are supported. For information about the conditions to be used with an 802.1p action, see [“Condition Combinations” on page 39-7](#) and [“Action Combinations” on page 39-9](#).

Note. 802.1p mapping can 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 AOS Release 6 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 39-67](#).

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 AOS Release 6 CLI Reference Guide*.

Note. A policy rule can 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 can only be modified through the source that created it. For example, if an action was created in PolicyView, it can 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 39-64](#).

To view information about how the switch classifies 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 39-48](#).

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 the indicated parameter can 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
r1      From Prec Enab  Act Refl Log Trap Save
(L2/3): cli    0  Yes  Yes  No  No  Yes  Yes
        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 the indicated parameter can 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 39-47](#).

An example of how the example configuration commands 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 (through the **snapshot** command), the commands included in the file includes syntax indicating the origin of the commands. 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 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 39-4.

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 39-64](#).

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 39-50](#) for more information about setting up groups.

More than one-condition parameter can be specified. Some condition parameters are mutually exclusive. For supported combinations of condition parameters, see [“Condition Combinations” on page 39-7](#).

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) is 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 displays an error message. For example:

```
ERROR: c3 is being used by rule 'my_rule'
```

In this case, the condition is not deleted. The condition (**c3**) must first be removed from the policy rule (**my_rule**). See [“Creating Policy Rules” on page 39-39](#) for more information about setting up rules.

If **c3** is not used by a policy rule, it is 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 must specify the way traffic must be treated. For example, it can specify a priority for the flow, a source address to rewrite in the IP header, or it can specify that the flow can 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 can 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 39-64](#).

More than one-action parameter can be specified. Some parameters are 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 39-7](#) and [“Action Combinations” on page 39-9](#). See the *OmniSwitch AOS Release 6 CLI Reference Guide* for details about command syntax.

policy action keywords (ingress and egress)

| | |
|-----------------------------|-------------------------|
| disposition | 802.1p |
| shared | dscp |
| priority | map |
| permanent gateway ip | port-disable |
| maximum bandwidth | redirect port |
| maximum depth | redirect linkagg |
| cir cbs pir pbs | no-cache |
| tos | mirror |

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 is to allow the flow classified by the policy condition.

The specified parameter (in this case, priority) is 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 displays an error message. For example:

```
ERROR: a6 is being used by rule 'my_rule'
```

In this case, the action is not deleted. The action (**a6**) must first be removed from the policy rule (**my_rule**). See [“Creating Policy Rules” on page 39-39](#) for more information about setting up rules.

If **a6** is not used by a policy rule, it is 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 is 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**) only takes effect after the **qos apply** command is entered. For more information about the **qos apply** command, see [“Applying the Configuration” on page 39-64](#).

The **policy rule** command specifies the following keywords:

policy rule keywords

precedence
validity period
save
log
log interval
count
trap

In addition, use the **policy rule** command to administratively disable or re-enable the policy rule. By default, rules are enabled. For a list of rule defaults, see [“Policy Rule Defaults” on page 39-13](#).

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 ensures that the rule can be enforced when the validity period becomes active.

Disabling Rules

By default, rules are enabled. Use the **policy rule** command to disable or re-enable the rules. For example:

```
-> policy rule rule5 disable
```

This command prevents **rule5** from being used to classify traffic.

If **qos disable** is entered, the rule is not used to classify traffic even if the rule is enabled. For more information about enabling/disabling QoS globally, see [“Enabling/Disabling QoS” on page 39-15](#).

Rule Precedence

The switch attempts to classify flows coming into the switch according to policy precedence. Only the rule with the highest precedence is applied to the flow. This is true even if the flow matches more than one rule.

Precedence is important for Access Control Lists (ACLs). For more details about precedence and examples for using precedence, see [Chapter 40, “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 is configurable 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 takes 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 is 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 can activate the rule for the current session, but the rule is not saved over a reboot. 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 AOS Release 6 Switch Management Guide* and the *OmniSwitch AOS Release 6 CLI Reference Guide*.

For more information about applying rules, see [“Applying the Configuration” on page 39-64](#).

Logging Rules

Logging a rule is useful for determining the source of firewall attacks.

To specify that the switch must 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
```

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 is deleted after the next **qos apply**.

Creating Policy Lists

A QoS policy list provides a method for grouping multiple policy rules 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.
- **User Network Profile (UNP)**—This type of configurable policy list is associated with an Access Guardian UNP. The rules in this list are applied to ingress traffic that is classified by the user profile. See Chapter 35, “Configuring Access Guardian,” for more information.
- **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 exist in the switch configuration. See [“Creating Policies” on page 39-34](#).
- 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 39-44](#).
- A rule can belong to the default list and an egress policy list at the same time. In addition, a rule can 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 39-44](#) for more information.
- Only one policy list per UNP is allowed, but a single policy list can be associated with multiple profiles. See [Chapter 35, “Configuring Access Guardian,”](#) 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 39-45](#) 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 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 39-7](#) and [“Action Combinations” on page 39-9](#) 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 as 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.
- Accounting mode is not supported for egress policy list. So if accounting mode is enabled for the rule then that rule can't be part of egress policy list and vice versa.
- 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 applies 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 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 39-69 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 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          cli  0Yes  Yes  No   No   Yes  Yes
{L2/3}:          cond5 -> action2

+my_rule5        cli  0Yes  No   No   No   Yes  Yes
{L2/3}:          cond2 -> pri2

mac1             cli  0Yes  No   No   No   Yes  Yes
{L2/3}:          dmac1 -> pri2
```

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 is not 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           cli  0      Yes Yes  No  No Yes Yes  0
{L2/3}:        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. 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 AOS Release 6 CLI Reference Guide* for more information about the output of these commands.

Testing Conditions

Before applying policies to the configuration, to see how the policies are used to classify traffic or how theoretical traffic is classified by policies that are already applied on the switch, use **qos apply** command.

Use the **show policy classify** commands to see how the switch classifies 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 must 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 must 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 displays the 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 the type 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 displays the 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 39-64](#).

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, 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 40, “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 40, “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 39-37.](#)

```
-> 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 AOS Release 6 CLI Reference Guide* for more information about the output of this display. See [“Verifying Condition Group Configuration” on page 39-59](#) 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 39-39.](#)) 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 39-64](#) 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. IPv6 addresses are not supported with network groups at this time. The policy condition specifies whether the network group is a source network group, destination network group, or multicast network group.

- **Default switch group**— By default, the switch contains a network group called **switch** that includes all IPv4 addresses configured for the switch itself. This network group can also be used in policy conditions.
- **ACLs**—Typically network groups are used for Access Control Lists. For more information about ACLs, see [Chapter 40, “Configuring ACLs.”](#)

To create a network policy group, use the **policy network group** command. Specify the name of the group and the IPv4 addresses to be included in the group. Each IPv4 address must be separated by a space. A mask can 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 is 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 can 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 addresses. 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 displays 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 39-37 for more information about configuring policy conditions.) The network group is 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 can be associated with policy service groups, which are then associated with policy conditions; or they can 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 can be used in combination.</i> |
| Shortcut for UDP service | source udp port destination udp port | <i>Keywords can 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 can be associated with a policy service group; or, individual services can 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 39-53](#).

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 services.

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 39-52](#) 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 can 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 is included in the condition. (For more information about configuring conditions, see [“Creating Policy Conditions” on page 39-37](#).)

Note. Service group configuration must be 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** is deleted at the next **qos apply**. If **serv_group** is associated with a policy condition, an error message is displayed 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 39-37](#) for more information about configuring policy conditions.) The service group is 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 is included in **macgrp2**.

The MAC group can then be associated with a condition through the **policy condition** command. The policy condition specifies whether the group must be used for *source* or *destination*. For example:

```
-> policy condition cond3 source mac group macgrp2
```

This command creates a condition called **cond3** that can 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 39-37](#).

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 addresses:

```
-> policy mac group macgrp2 no 08:00:20:00:00:00
```

This command specifies that MAC address 08:00:20:00:00:00 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** is deleted at the next **qos apply**. If **macgrp2** is associated with a policy condition, an error message is displayed 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 39-37](#) for more information about configuring policy conditions. The MAC group is deleted at the next **qos apply**.

Creating Port Groups

Port groups are made up of slot and port number combinations. 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**, and so on). 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 can then be associated with a condition through the **policy condition** command. The policy condition specifies whether the group must be used for *source* or *destination*. For example:

```
-> policy condition cond4 source port group techpubs
```

This command creates a condition called **cond4** that can 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 39-37](#).

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 numbers.

```
-> policy port group techpubs no 2/1
```

This command specifies that port 2/1 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** is deleted at the next **qos apply**. If **techpubs** is associated with a policy condition, an error message is displayed 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 39-37](#) for more information about configuring policy conditions.) The port group is deleted at the next **qos apply**.

Port Group and Per Port Rate Limiting

Per port rate limiting allows configuring a policy rule that specifies a rate limiter for the group of ports or individual port. This can be achieved by configuring specific mode for the port group. The following two modes are supported:

- **Non-split:** This mode applies the rate limiting rule to a group of ports specified in the rule. This is the default behavior for the source port group.
- **Split:** This mode applies the rate limiting rule to an individual port specified in the group of ports in the rule.

Per port rate limiting is limited to the source port group attached to the default policy list. The configuration is not valid for any other policy list. Hence, the configuration of the policy rule for the split mode is not valid for the explicit policy lists including ingress.

Rate limiting action can be applied as a part of the rule to each port. Actions such as DSCP value, priority, and so on can also be applied in addition to the rate limiting. Policy action ‘shared’ cannot be used with the rule where split source port group is configured. Shared policy action of the meter is applicable across the rules that share the action. Since multiple meters are used corresponding to each port configured with the source port group in the split mode in the rule, shared action cannot be used.

To configure rate limiting to split mode in a defined port group, use the following command. For example:

```
-> policy port group techpubs mode split 1/1-2
```

To configure rate limiting to non-split mode in a defined port group, use the following command. For example:

```
-> policy port group techpubs mode non-split 1/1-2
```

Note. Rate limiting is not supported for destination port group, and an error is displayed at the time of rule creation for the destination port group condition.

Port Groups and Maximum Bandwidth

Maximum bandwidth policies are applied to source (ingress) ports and flows. This applies to flows that involve more than one port (port group). Based on the rate limit mode set on the port group, the maximum bandwidth is applied to ports individually or together. For example,

- If a policy specifies a maximum bandwidth value of 10M for a port group with rate limiter set as non-split mode containing 4 ports on the same slot, the total bandwidth limit enforced is 10M for all 4 ports.
- If a policy specifies a maximum bandwidth value of 10M for a port group with rate limiter set as split mode containing 4 ports, then bandwidth of 10M is applied to each of the 4 ports, that is, a total of 40M bandwidth is enforced.
- If a policy specifies a maximum bandwidth value of 10M for a port group with rate limiter set as split mode containing 4 ports on the same slot, and 2 ports on different slots, then bandwidth of 10M is applied to each of the 4 ports in the same slot, and also 10M each for the ports located on different slots.
- If a policy specifies a maximum bandwidth value of 10M for a port group with rate limiter set as non-split mode containing 4 ports on the same slot, and 2 ports on different slots, then bandwidth of 10M is shared across all the 4 ports in the same slot, and a bandwidth of 10M is shared to the ports located on different slots.

Following are some points to note while configuring ingress maximum bandwidth policies:

- The **show active policy rule** command displays the number of packets that were dropped (match counter) because they exceeded the ingress bandwidth limit applied by a maximum bandwidth policy.
- Although bandwidth policies are applied to ingress ports, it is possible to specify a destination port or destination port group in a bandwidth policy as well. Doing so affects the egress rate limiting/egress policing on the ingress port itself.

The following subsections provide examples of ingress maximum bandwidth policies using both source and destination port groups.

Example 1: Source port group with non-split mode or default mode

In the following example, a port group (**pgroup**) is created with two ports and rate limiter set as non-split mode or default mode, and attached to a policy condition (**Ports**). A policy action (**MaxBw**) is created with maximum bandwidth of 10k. 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 10k maximum bandwidth is shared by both ports.

Example 2: Source port group with split mode

In the following example, a port group (**pgroup**) is created with two ports and rate limiter set as split mode, and attached to a policy condition (**Ports**). A policy action (**MaxBw**) is created with maximum bandwidth of 10k. The policy condition and policy action are combined in a policy rule called **PortRule**.

```
-> policy port group pgroup mode split 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 maximum bandwidth of the 10k is applied to both the ports separately.

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 can 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 can 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 39-37](#).

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 addresses:

```
-> policy mac group vlangrp1 no 15
```

This command specifies that VLAN ID 15 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 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** is deleted at the next **qos apply**. If **vlangrp1** is associated with a policy condition, an error message is displayed 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 39-37](#) for more information about configuring policy conditions. The MAC group is 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 AOS Release 6 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 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 (+) is removed from **netgroup1** in the display. See [“Applying the Configuration” on page 39-64](#) 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 39-62](#) and [“Creating Map Groups” on page 39-62](#).

```
-> 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 39-38](#) 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 39-63](#) and the *OmniSwitch AOS Release 6 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 39-39](#).

```
-> policy rule r3 condition Traffic action tosMap
```

4 Apply the configuration. For more information about this command, see [“Applying the Configuration” on page 39-64](#).

```
-> qos apply
```

How Map Groups Work

When mapping from 802.1p to 802.1p, the action results 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 are remapped according to **Group2**. If the incoming 802.1p value is 1 or 2, the value is 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 is mapped to 5. If the incoming 802.1p value is 5 or 6, the value is 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 are not included in the map group are 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 is mapped according to **Group2** if the value is specified in **Group2**. If the incoming ToS value is 2, the value is mapped to 5; however, if the incoming value is 3, the switch maps the value to zero as 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 is 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 is mapped to 5.

Note. Map group configuration is not active until the **qos apply** command is entered.

The remapping group can 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 are 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 is displayed:

```
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 is 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 AOS Release 6 CLI Reference Guide*.

When the command is used to show output for all pending and applied condition groups, the following characters 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 (+) is removed from **tosGroup** in the display. See [“Applying the Configuration” on page 39-64](#) for more information about the **qos apply** command.

Applying the Configuration

Configuration for policy rules and many global QoS parameters must 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 must 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 become active immediately.* Other global commands must 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 can be used to review information about new conditions before they are applied on the switch. See [“Testing Conditions” on page 39-48](#).

Applied policy rules can also be administratively disabled (inactive). If a rule is administratively disabled, the rule exists in the applied configuration but not used to classify flows. For more information about disabling/re-enabling a policy rule, see [“Creating Policy Rules” on page 39-39](#).

Deleting the Pending Configuration

Policy settings that have been configured but not applied through the **qos apply** command can 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 then looks like:

| Pending Policies | Applied Policies |
|------------------|------------------|
| rule1 | rule1 |
| rule2 | rule2 |
| rule3 | rule3 |

Flushing the Configuration

In some cases, you can remove all of your rules and start over again. To erase pending policies from the configuration completely, use the **qos flush** command. For example:

```
-> qos flush
```

If you then enter **qos apply**, all policy information is 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 then looks 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 are written to the applied policies and all policy rule configuration is deleted.

Interaction With LDAP Policies

The **qos apply**, **qos revert**, and **qos flush** commands do not affect policies created through the PolicyView application. Separate commands are used for loading and flushing LDAP policies on the switch. See [Chapter 36, “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 handles the packet. Use the applied keyword to examine only applied conditions. |

For more information about these commands, see the *OmniSwitch AOS Release 6 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 can be as simple as identifying a Layer 2 or Layer 3 address of an incoming flow. Treating the traffic can 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 |

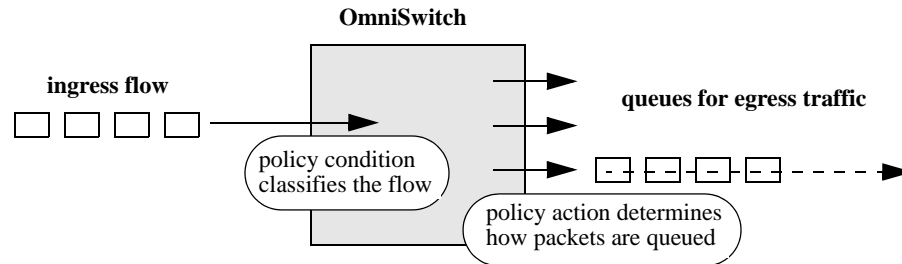
Note. The redirection policies, policy based mirroring, and policy based routing (PBR) are not supported by egress policies.

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 40, “Configuring ACLs.”](#)

Policies can also be used for prioritizing traffic in dynamic link aggregation groups. For more information about dynamic link aggregates, see [Chapter 26, “Configuring Dynamic Link Aggregation.”](#)

Basic QoS Policies

Traffic prioritization and bandwidth shaping is the most common type of QoS policies. For these policies, any condition can be created; the policy action indicates how the traffic must be prioritized or how the bandwidth must be shaped.



Basic QoS Policy Application

Note. To configure same priority for multiple addresses, services, or ports, use a policy condition group to specify the group and associate the group with the condition. See [“Using Condition Groups in Policies”](#) on page 39-50 for more information about groups.

Some condition parameters can 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 39-50 and [“Condition Combinations”](#) on page 39-7.

Basic Commands

The following **policy action** commands are used for traffic prioritization or shaping:

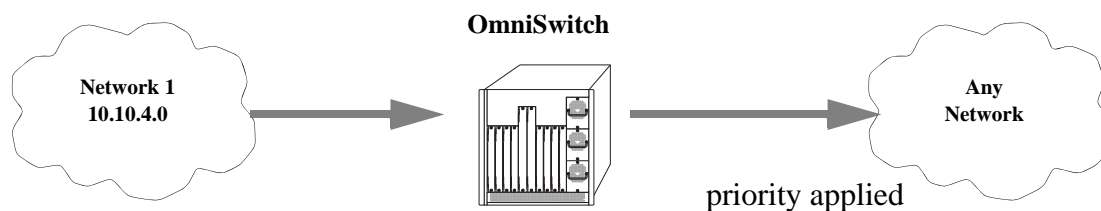
policy action priority
policy action maximum bandwidth

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 AOS Release 6 CLI Reference Guide*.

QoS ports can 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 is given the highest priority.

Bandwidth Shaping Example

In this example, a specific flow from a source IP address is sent to a queue that supports 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
```

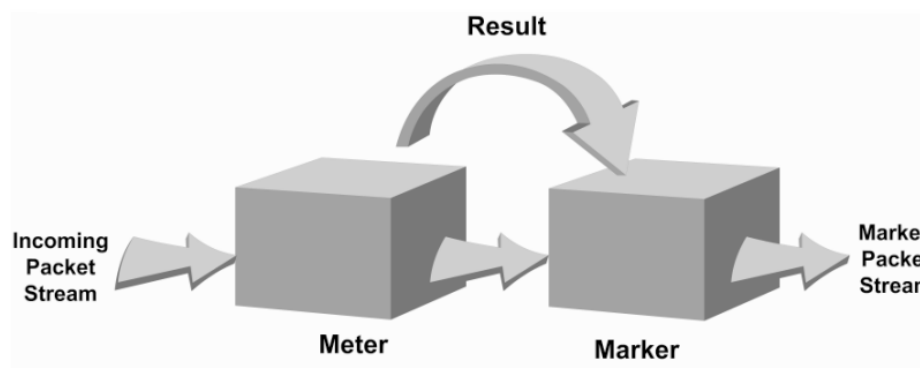
The bandwidth can be specified in abbreviated units, in this case, **1k**.

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 is 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 functions 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 39-70](#) 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 not CIR/CBS or 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 not CIR/CBS or 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)
- **counter-color** (packet colors to count for TCM statistics)

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 **show policy classify** 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
```

The policy action has a **no** command to remove the CIR, CBS, PIR, PBS information. 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 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
```

The rates and burst sizes can 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 are 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 can 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
```

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.

Note the following regarding the use and configuration of redirection policies:

- Redirection policies apply to both bridged and routed traffic.
- When redirecting routed traffic from VLAN A to VLAN B, the redirect port or link aggregate ID must belong to VLAN A (tagged or default VLAN).
- Routed packets (from VLAN A to VLAN B) are not modified after they are redirected; the source and MAC address remain the same. In addition, if the redirect port or link aggregate ID is tagged, the redirected packets have a tag from the ingress VLAN A.
- If a route exists for the redirected flow, then redirected packets are the final post-routing packets.
- If a route does not exist for the redirected flow, the flow is not redirected to the specified port or link aggregate ID and is “blackholed”. As soon as a route is available, the flow is then redirected as specified in the policy.

In most cases, a redirected flow will not trigger an update to the routing and ARP tables. When the ARP table is cleared or timed out, port/link aggregate redirection will cease until the ARP table is refreshed. If necessary, create a static route for the flow or assign the redirect port or link aggregate ID to the ingress VLAN (VLAN A) to send packets to the redirect port until a route is available.

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
```

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 can 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, flows coming into the switch from source IP address 192.168.20.1 are mirrored to port 1/10. For example:

```
-> policy condition c1 source ip 192.168.20.1
-> policy action a1 mirror ingress 1/10 disposition deny
-> policy rule r1 condition c1 action a1
-> qos apply
```

This policy rule example combines the MTP action with the deny action. As a result, this rule denies 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 must 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

You can configure policies for ICMP on a global basis on the switch. ICMP policies can 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

You can map the 802.1p values to different 802.1p values on an individual basis or by using a map group. In addition, ToS or DSCP values can be mapped to 802.1p on a case-by-case basis or through a map group. (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:

- Ingress ports can be legacy ports or UNI ports
- Priority for the flow is based on the policy action. The value specified for 802.1p, ToS, DSCP, or the map group 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 39-29](#).

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 must be mapped; the **policy action map** command specifies what must be mapped (802.1p to 802.1p, ToS/DSCP to 802.1p) and the mapping group that must be used. For more details about creating map groups, see [“Creating Map Groups” on page 39-62](#).

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

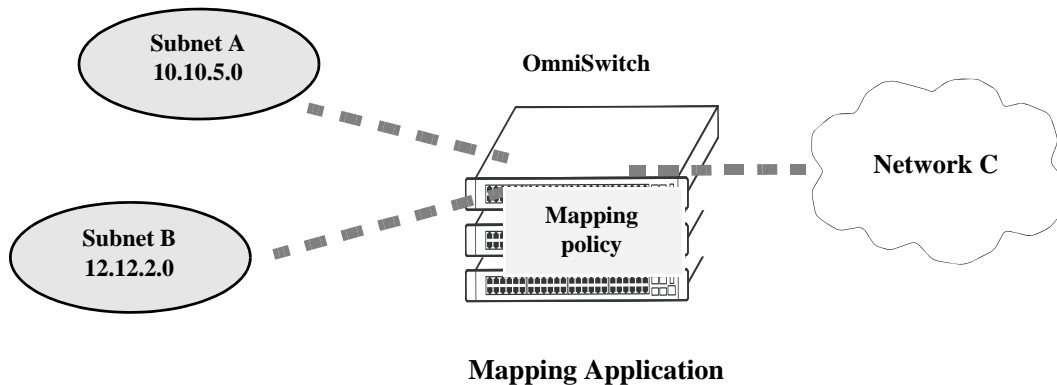
```

The **map_action** specifies that ToS values are 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 overrides the normal routing mechanism for traffic matching the policy condition.

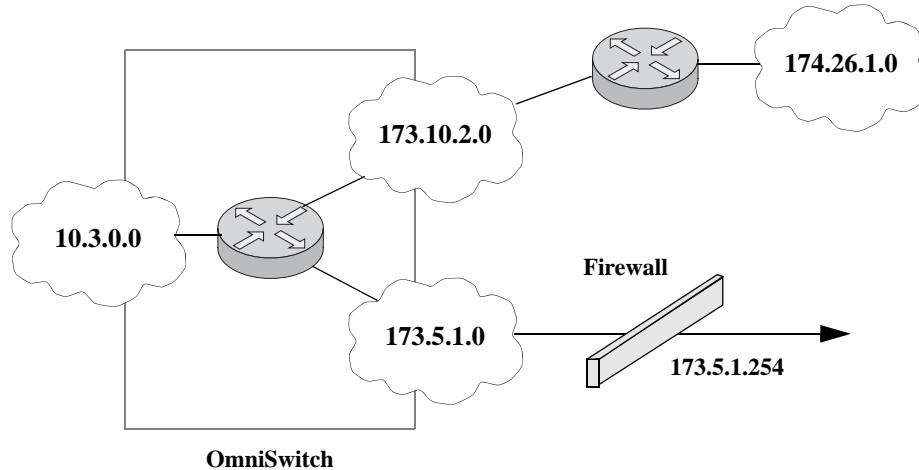
Note. When a PBR QoS rule is applied to the configuration, it is applied to the entire switch, unless you specify a built-in port group in the policy condition.

Policy-Based Routing can 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 can be redirected to a particular gateway regardless of what routes are listed in the routing table. The gateway address does not have to be on a directly connected VLAN; the address can 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 is redirected to the route specified in the policy. For information about viewing the routing table, see [Chapter 28, “Configuring IP.”](#)

Policy-Based Routing can be used to redirect untrusted traffic to a firewall. In this case, reply packets are not allowed back through the firewall.



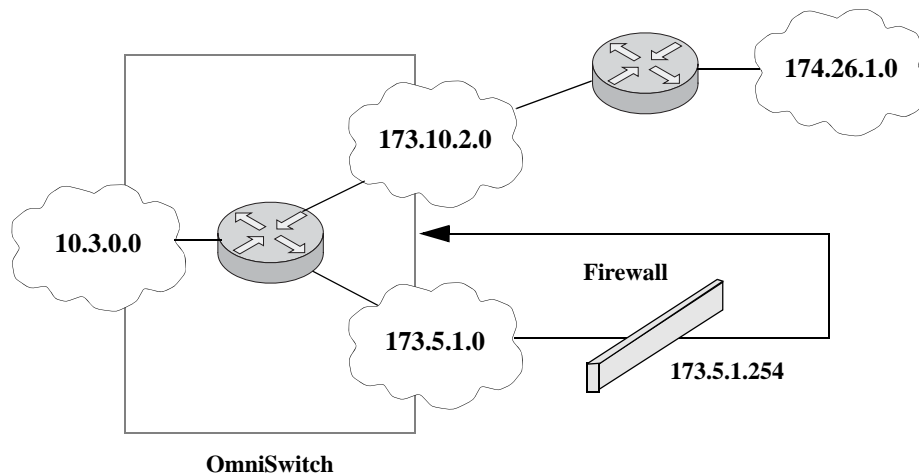
Routing all IP source traffic through a firewall

In this example, all traffic originating in the 10.3 network is routed through the firewall, regardless of whether 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
```

The functionality of the firewall is important. In the example, the firewall is sending the traffic to be routed remotely. Instead, if you set up a firewall to send the traffic back to the switch to be routed, set up the policy condition with a built-in source port group so that traffic coming back from the firewall does not get looped and is sent back 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 rerouted. 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
```

Ensure to use the **qos apply** command to activate the policy rule on the switch. Otherwise, the rule is saved as a part of the pending configuration. However this rule is not activated.

40 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 39, “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 AOS Release 6 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 40-7.](#)
- **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 40-8.](#)
- **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 40-8.](#)
- **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 40-15.](#)

Note. ACLs may also be created on an external LDAP server through a network management application such as PolicyView and downloaded to the switch. For information about managing rules on an LDAP server, see [Chapter 2, “Managing Policy Servers.”](#)

ACL Specifications

The QoS/ACL functionality described in this chapter is supported on the OmniSwitch 6350 and OmniSwitch 6450. Note that any maximum limits provided in the Specifications table are subject to available system resources.

| | |
|--|--|
| Platforms Supported | OmniSwitch 6350, 6450 |
| Maximum number of policy rules | 1024 (ingress and egress rules combined) |
| Maximum number of egress policy rules | 512 |
| 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 |
| 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 AOS Release 6 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) |
| Policy rule accounting | policy rule accounting | disable |

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 39, “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 13 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 39-48 in Chapter 39, “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
```

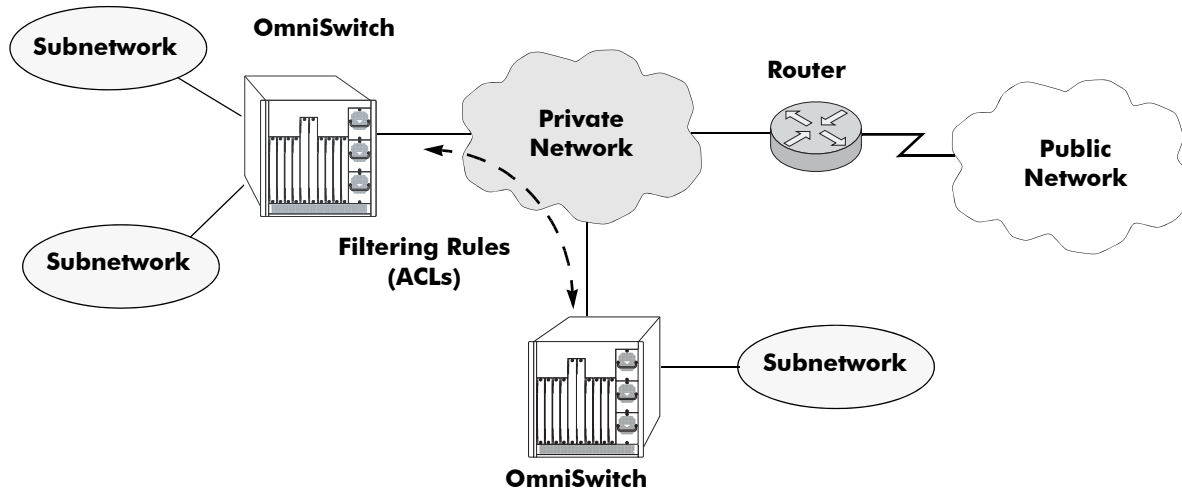
5 Create a policy rule with the **policy rule** command and include the relevant condition and action. Use the keyword **precedence** to specify the priority of this rule over other rules for traffic matching the specified condition.

```
-> 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 39-64 in Chapter 39, “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 40-7 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 40-6 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 39, “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 39-22 in Chapter 39, “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 39-7](#) and [“Action Combinations” on page 39-9 in Chapter 39, “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 39, "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 40-7.
- 2 Create a condition for the traffic to be filtered.** This step is described in ["Creating Condition Groups For ACLs"](#) on page 40-8 and ["Creating Policy Conditions For ACLs"](#) on page 40-9.
- 3 Create an action to accept or deny the traffic.** This step is described in ["Creating Policy Actions For ACLs"](#) on page 40-10.
- 4 Create a policy rule that combines the condition and the action.** This step is described in ["Creating Policy Rules for ACLs"](#) on page 40-10.

For a quick tutorial on how to configure ACLs, see ["Quick Steps for Creating ACLs"](#) on page 40-4.

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'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 39, "Configuring QoS."](#) and the *OmniSwitch AOS Release 6 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 40-11](#).

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 39-50 in Chapter 39, “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 39, “Configuring QoS.”](#) Command syntax is described in detail in the *OmniSwitch AOS Release 6 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 40-8](#).

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 mac group destination mac destination mac group source vlan source vlan group source port source port group ethertype 802.1p | source ip source ipv6 source network group destination ip destination ipv6 destination network group source ip port destination ip port service service group ip protocol ipv6 icmptype icmpcode tos dscp source tcp port destination tcp port source udp port destination udp port established tcpflags | multicast ip multicast ipv6 multicast network group destination ip destination vlan destination port destination port group destination mac destination mac group |

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 40-6](#) 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 39-64](#) in Chapter 39, “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 39-7](#) and [“Action Combinations” on page 39-9](#) in [Chapter 39, “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 39-7](#) and [“Action Combinations” on page 39-9](#) in [Chapter 39, “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 AOS Release 6 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 41, “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 40-16](#).
- **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 40-17](#).
- **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 **icmpcode**. See [“Configuring ICMP Drop Rules” on page 40-18](#).
- **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 40-19](#).
- **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 ether type (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 40-17](#) 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 AOS Release 6 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 AOS Release 6 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
```

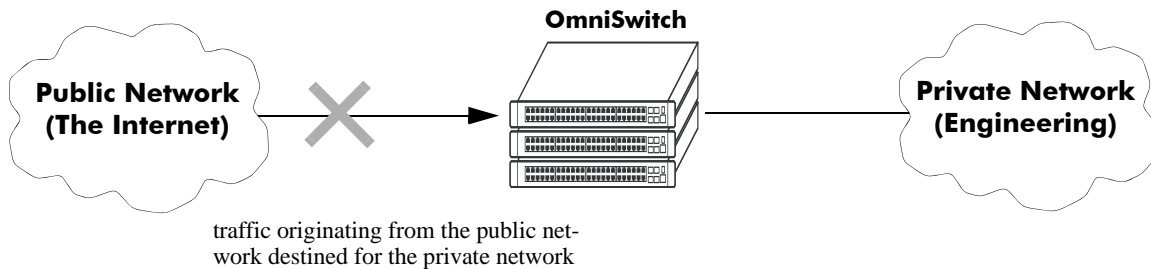
| | Policy | From | Prec | Enab | Inact | Refl | Log | Save | Matches |
|-----------|--------|-------|------|------|-------|------|-----|------|---------|
| +my_rule5 | | cli | 0 | Yes | No | No | No | Yes | 0 |
| Cnd/Act: | | cond2 | -> | pri2 | | | | | |
| mac1 | | cli | 0 | Yes | No | No | No | Yes | 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 AOS Release 6 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
```

41 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 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 AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Enabling and disabling IP Multicast Switching on [page 41-8](#).
- Enabling and disabling IP Multicast Dynamic Control on [page 41-9](#)
- Configuring and removing an IGMP static neighbor on [page 41-11](#).
- Enabling and Disabling Static Neighbor Fast Convergence on [page 41-12](#).
- Configuring and removing an IGMP static querier on [page 41-12](#).
- Configuring and removing an IGMP static group on [page 41-14](#).
- Modifying IPMS parameters beginning on [page 41-15](#).
- Enabling and disabling IPv6 Multicast Switching on [page 41-25](#).
- Configuring and removing an MLD static neighbor on [page 41-27](#).
- Configuring and removing an MLD static querier on [page 41-28](#).

- Configuring and removing an MLD static group on [page 41-28](#).
- Modifying IPMSv6 parameters beginning on [page 41-30](#).
- Star-G mode on Multicast Group [page 41-38](#).

Note. You can also configure and monitor IPMS with WebView, Alcatel 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 online documentation for more information on configuring and monitoring IPMS/IPMSv6 with WebView.

IPMS Specifications

The table below lists specifications for Alcatel 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 6350, 6450 |
| 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 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 6350, 6450 |
| 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 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 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 |

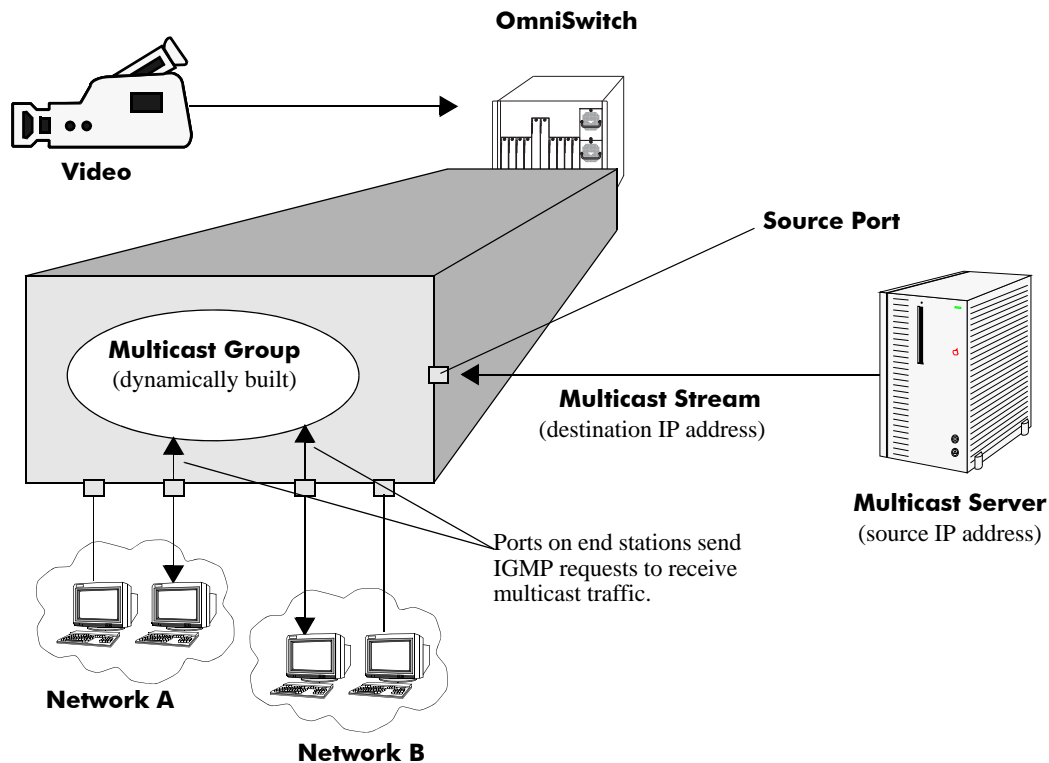
IPMS Overview

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 41-7 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 (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 (for example, RIP2) |
| 224.0.1.0 through 224.0.1.255 | Internetwork Control Block (for example, RSVP, DHCP, commercial servers) |
| 224.0.2.0 through 224.0.255.0 | AD-HOC Block (for example, 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 41-8), configure a port as a IGMP static neighbor (see “[Configuring and Removing an IGMP Static Neighbor](#)” on page 41-11), configure a port as a IGMP static querier (see “[Configuring and Removing an IGMP Static Querier](#)” on page 41-12), and configure a port as a IGMP static group (see “[Configuring and Removing an IGMP Static Group](#)” on page 41-14).

In addition, a tutorial is provided in “[IPMS Application Example](#)” on page 41-40 that shows how to use CLI commands to configure a sample network.

Note. See the “IP Multicast Switching Commands” chapter in the *OmniSwitch AOS Release 6 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 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 (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 (disabled).

Enabling and Disabling IP Multicast Dynamic Control

IP multicast dynamic control avoids high CPU usage in the network. The CPU usage increase if there is a high rate of packets being captured to the CPU. This feature controls the processing of IPV4 protocol packets in the CPU.

On enabling the IP multicast dynamic control on the switch, the IPV4 protocol packets entering the switch are transparently forwarded without any CPU processing. The packets are not captured to the CPU.

Enabling IP Multicast Dynamic Control

To enable the IP multicast dynamic control, use the [ip multicast dynamic-control drop-all status](#) command as shown below:

```
-> ip multicast dynamic-control drop-all status enable
```

Note. This feature should not be enabled if routing protocol or VRRP is configured on the switch.

Disabling IP Multicast Dynamic Control

To disable the IP multicast dynamic control, use the [ip multicast dynamic-control drop-all status](#) command as shown below:

```
-> ip multicast dynamic-control drop-all status disable
```

Note. To check the status, use the [show ip multicast](#) command.

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, 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, 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 (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 (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 (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.

Note. The IGMS implementation frame (join/leave report & query) now supports multiple tagging of up to 8 VLAN tags in the frame.

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 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 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, enter:

```
-> no ip multicast static-neighbor vlan 2 port 4/10
```

Enabling and Disabling Static Neighbor Fast Convergence

You can enable fast convergence for multicast switching traffic over DHL and ERIPv2. It is applicable only for IPMS static neighbors, and not applicable to forward entries created by IGMP group packets.

IP multicast fast convergence works only in standalone mode.

Enabling the Static Neighbor Fast Convergence

To enable IP multicast static neighbor fast convergence, use the **enable** keyword in the **ip multicast static-neighbor fast-convergence** command as shown below:

```
-> ip multicast static-neighbor fast-convergence enable
```

Disabling the Static Neighbor Fast Convergence

To disable IP multicast static neighbor fast convergence, use the **disable** keyword in the **ip multicast static-neighbor fast-convergence** command as shown below:

```
-> ip multicast static-neighbor fast-convergence disable
```

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, 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, 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, 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, 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, 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, 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 41-4 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 (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, 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 (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 (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, 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 (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 (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, 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 (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 van 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 (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, 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 (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 (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, 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 (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, 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, enter:

```
-> ip multicast querying disable
```

Or, as an alternative, enter:

```
-> ip multicast querying
```

To restore the IGMP querying to its default setting (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 (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 (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, 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 (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 (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 (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 (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 (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 (processing membership and source filter removals immediately without waiting for the protocol 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 (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 (disabled).

Limiting IGMP Multicast Groups

By default there is no limit on the number of IGMP groups that can be learned on a port/vlan instance. A maximum group limit can be set on a port, VLAN or on a global level to limit the number of IGMP groups

that can be learned. Once the configured limit is reached, a configurable action decides whether the new IGMP report is dropped or replaces an existing IGMP membership.

The maximum group limit can be applied globally, per VLAN, or per port. Port settings override VLAN settings, which override global settings.

If the maximum number of groups is reached an action can be configured to either drop the new membership request or replace an existing group membership as show below.

Setting the IGMP Group Limit

To set the IGMP global group limit and drop any requests above the limit, use the **ip multicast max-group** command as shown below:

```
-> ip multicast max-group 25 action drop
```

To set the IGMP group limit for a VLAN and replace an existing session use the **ip multicast vlan max-group** command as shown below:

```
-> ip multicast vlan 10 max-group 25 action replace
```

To set the IGMP group limit for a port and drop any requests above the limit, use the **ip multicast port max-group** command as shown below:

```
-> ip multicast port 1/1 max-group 25 action drop
```

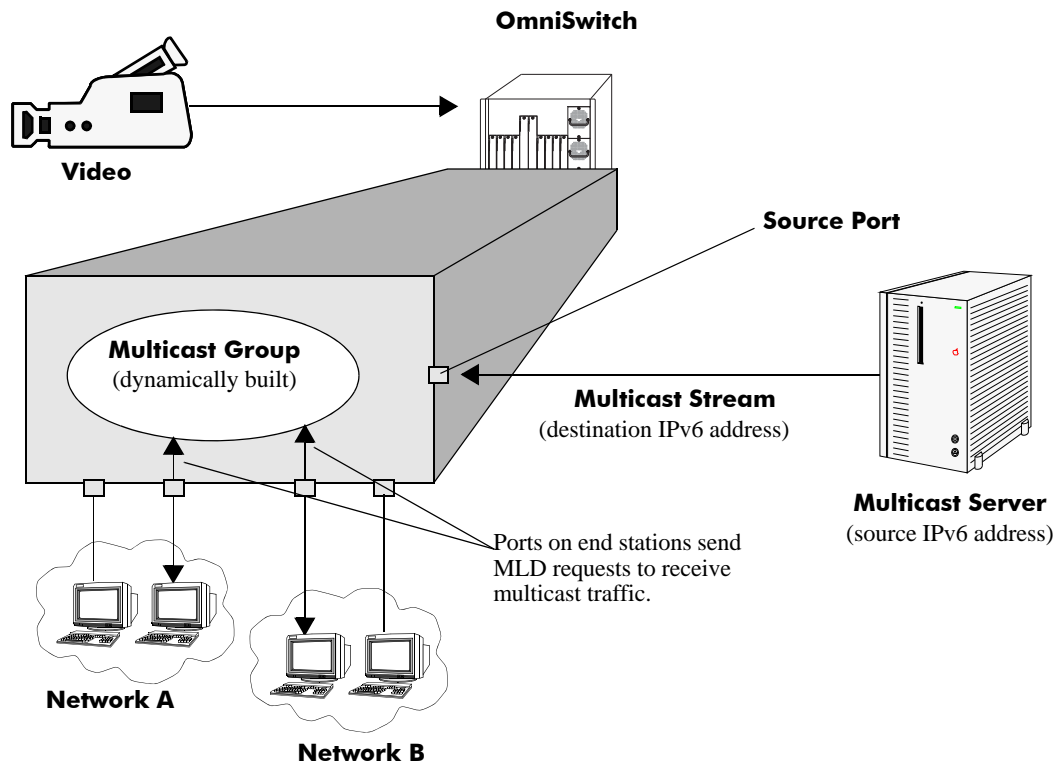
IPMSv6 Overview

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 41-23.)

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 (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 |

| Address | Description |
|--------------------|--------------------------|
| 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 41-26 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 41-25](#)), configure a port as an MLD static neighbor (see [“Configuring and Removing an MLD Static Neighbor” on page 41-27](#)), configure a port as an MLD static querier (see [“Configuring and Removing an MLD Static Querier” on page 41-28](#)), and configure a port as an MLD static group (see [“Configuring and Removing an MLD Static Group” on page 41-28](#))

Note. See the “IP Multicast Switching Commands” chapter in the *OmniSwitch AOS Release 6 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 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, 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, 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 (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 (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, 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, 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, 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, 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, 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, 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, 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, enter:

```
-> no ipv6 multicast static-group ff05::5 vlan 3 port 3/10
```

Modifying IPMSv6 Parameters

The table in “[IPMSv6 Default Values](#)” on page 41-5 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 (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, 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 (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 (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, 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 (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 (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 (1000 milliseconds) value.

Modifying the MLD Query Response Interval

The default MLD query response interval (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, 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 (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 (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, 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 (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 (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, 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 (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, 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, enter:

```
-> ipv6 multicast querying disable
```

Or, as an alternative, enter:

```
-> ipv6 multicast querying
```

To restore the MLD querying to its default setting (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 (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 (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, 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 (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 (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 (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 (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 (processing membership and source filter removals immediately without waiting for the protocol 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 (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 (disabled).

Limiting MLD Multicast Groups

By default there is no limit on the number of MLD groups that can be learned on a port/vlan instance. A maximum group limit can be set on a port, VLAN or on a global level to limit the number of MLD groups that can be learned. Once the configured limit is reached, a configurable action decides whether the new MLD report is dropped or replaces an existing MLD membership.

The maximum group limit can be applied globally, per VLAN, or per port. Port settings override VLAN settings, which override global settings.

If the maximum number of groups is reached an action can be configured to either drop the new membership request or replace an existing group membership as show below.

Setting the MLD Group Limit

To set the MLD global group limit and drop any requests above the limit, use the **ipv6 multicast max-group** command as shown below:

```
-> ipv6 multicast max-group 25 action drop
```

To set the MLD group limit for a VLAN and replace any requests above the limit, use the **ipv6 multicast vlan max-group** command as shown below:

```
-> ipv6 multicast vlan 10 max-group 25 action replace
```

To set the MLD group limit for a port and drop any requests above the limit, use the **ipv6 multicast port max-group** command as shown below:

```
-> ipv6 multicast port 1/1 max-group 25 action drop
```

Star-G Mode for Multicast Group

When multiple hosts are a part of single multicast group, every host will have an unique entry in the IPMC table. This occupies more hardware entries in IPMC thus affecting other normal multicast services. In such a scenario, configuring star-G (*, G) mode for the multicast group reduces the IPMC index utilization by preventing creation of multiple multicast entries. Single star-G entry for the multicast group is created in the IPMC table.

A scenario where star-G can be implemented:

Universal Plug and Play (UPnP) is a set of networking protocols that permits network devices such as personal computers, printers, Internet gateways, Wi-Fi access points and mobile devices to seamlessly discover each other's presence on the network and establish functional network services for data sharing, communications, and entertainment. Windows PC using UPnP services acts as both server and client. For example:

- Server - The PC is multicasting the "services" on a reserved multicast group 239.255.255.20 (UDP port 1910)
- Client - The PC is subscribing to the UPnP service by joining the multicast group 239.255.255.250

In the above example, enabling star-G mode for multicast group 239.255.255.20 and 239.255.255.250 eliminates creation of IPMC entries from each host for the server and client. Instead, a single star-G entry for the multicast group is created.

Enabling and Disabling star-G Mode Globally

To enable or disable star-G mode (*, G) for IPv4 multicast switching, use the **ip multicast star-g-mode status** command. By default, star-G mode is disabled on the switch.

```
-> ip multicast star-g-mode status enable
-> ip multicast star-g-mode status disable
```

Note. IGMP v3 must not be enabled on the switch when star-G mode is in operation at the global and per VLAN level.

Enabling and Disabling star-G Mode on a VLAN

To enable or disable star-G mode (*, G) for IPv4 multicast switching on a specific VLAN, use the **ip multicast vlan star-g-mode status** command.

```
-> ip multicast vlan 10 star-g-mode status enable
-> ip multicast vlan 10 star-g-mode status disable
```

Note.

- IGMP v3 must not be enabled on the VLAN when star-G mode is in operation.
 - IP multicast must be enabled globally or at the VLAN level for star-G mode to be enabled for a specific VLAN.
-

Verifying star-G Mode Configuration

The following commands displays the star-G mode configuration:

show ip multicast Displays the IP Multicast Switching and Routing status and the general configuration parameters on the specified VLAN or on the system if no VLAN is specified.

show ip multicast source Displays the IP Multicast Switching and Routing source table entries matching the specified IP multicast group address or all entries if no IP multicast group address is specified.

show ip multicast forward Displays the IP Multicast Switching and Routing forwarding table entries for the specified IP multicast group address or all the entries if no IP multicast group address is specified.

show configuration snapshot ipms can also be used to view the star-G mode configuration details as shown below.

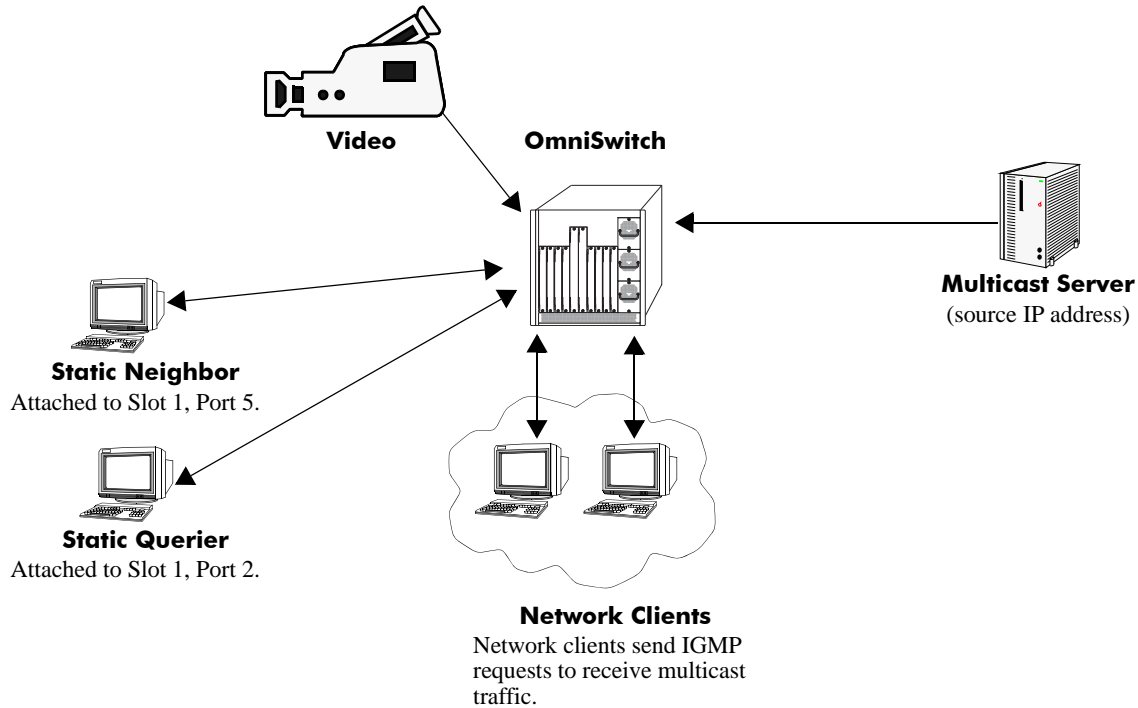
For example:

```
-> show configuration snapshot ipms
! IPMS :
ip multicast status enable
ip multicast querying enable
ip multicast vlan 20 star-g-mode 239.255.255.20
ip multicast vlan 20 star-g-mode 239.255.255.250
ip multicast vlan 20 status enable
ip multicast vlan 20 querying enable
```

Note. See the “IP Multicast Switching Commands” chapter in the *OmniSwitch AOS Release 6 CLI Reference Guide* for more information on the star-G mode **show** command.

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 (7) value.

Follow the steps below to configure this network:

Note. All the steps following Step 1 (which must be executed first) can 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:                              enabled,
Proxying:                              disabled,
Spoofing:                              disabled,
Zapping:                               disabled,
Querier Forwarding:                   disabled,
Flood Unknown:                        disabled,
Dynamic control status:                disabled,
Dynamic control drop-all status:      disabled,
Buffer Packet:                        disabled,
Version:                               2,
Robustness:                            7,
Query Interval (seconds):              125,
Query Response Interval (tenths of seconds): 100,
Last Member Query Interval (tenths of seconds): 10,
Unsolicited Report Interval (seconds):  1,
Router Timeout (seconds):              90,
Source Timeout (seconds):              30,
Max-group:                             0,
Max-group action:                      none,
Helper-address:                        0.0.0.0
```

```
-> show ip multicast neighbor
```

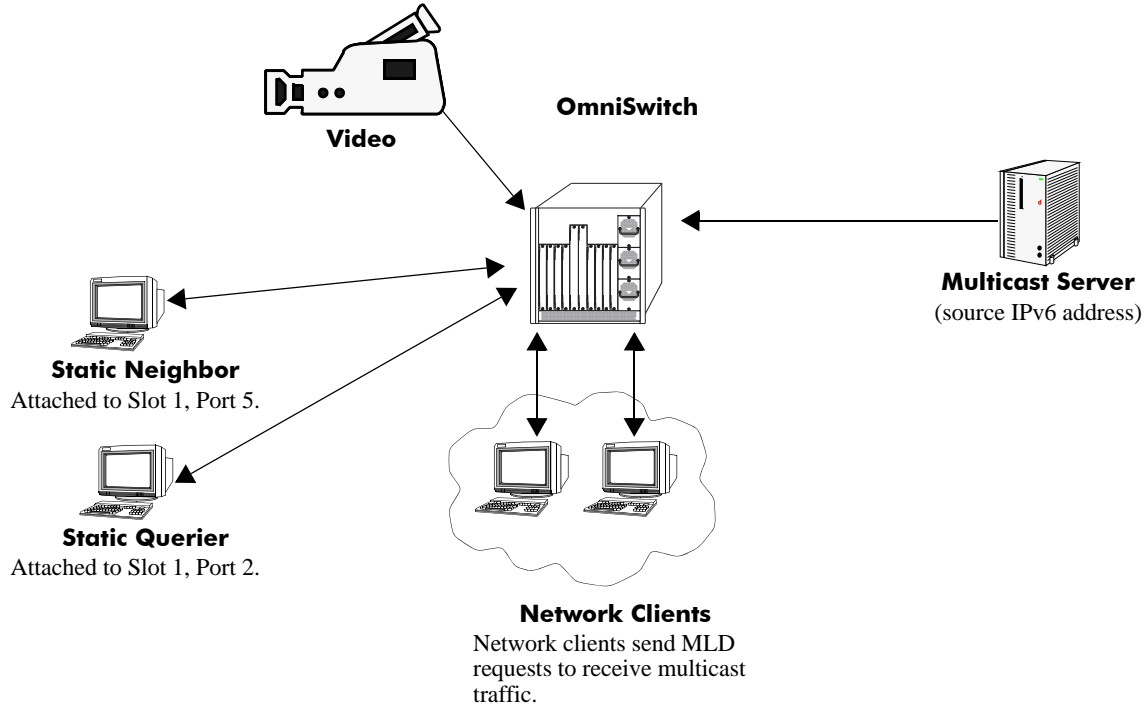
```
Total 1 Neighbors
Host Address   VLAN  Port  Static  Count  Life
-----+-----+-----+-----+-----+-----
1.0.0.2       5    1/5  no      1      86
```

```
-> 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 (7) value.

Follow the steps below to configure this network:

Note. All the steps following Step 1 (which must be executed first) can 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 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 AOS Release 6 CLI Reference Guide* for complete documentation on IPMS **show** commands.

Displaying IPMSv6 Configurations and Statistics

Alcatel 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 AOS Release 6 CLI Reference Guide* for complete documentation on IPMS **show** commands.

42 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. The bulk of data is transmitted 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 on command syntax, see the *OmniSwitch AOS Release 6 CLI Reference Guide*.

Configuration procedures described in this chapter include:

- Creating and Deleting IPMVLAN on [page 42-9](#).
- Assigning and Deleting IPv4/IPv6 Addresses on [page 42-10](#).
- Assigning and Deleting a C-Tag on [page 42-10](#).
- Creating and Deleting a Sender Port on [page 42-10](#).
- Creating and Deleting a Receiver Port on [page 42-11](#).
- Associating an IPMVLAN with a Customer VLAN on [page 42-12](#).

Note. You can also configure and monitor IPMV through WebView. It is the embedded web-based device management application of Alcatel. WebView is an interactive and easy-to-use GUI that can be launched from OmniVista or a web browser. See the online documentation on WebView 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 6350, 6450 |
| 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 has to 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 (that is, 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 has 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 only flows from the distribution VLAN to the customer VLAN. Customer-generated multicast traffic flows 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.

Multicast VLAN Registration

Multicast VLAN Registration (MVR) feature allows several subscribers from different VLANs on a trunk interface to subscribe and unsubscribe to a single multicast VLAN. Multicast VLAN Registration (MVR) is designed for applications using wide-scale deployment of multicast traffic across a service-provider network. MVR allows a subscriber on a port to subscribe and unsubscribe to a multicast stream on the network-wide multicast VLAN. It allows the single multicast VLAN to be shared in the network while subscribers remain in separate VLANs. MVR provides the ability to send continuous multicast streams in the multicast VLAN, but to isolate the streams from the subscriber VLANs for bandwidth and security reasons.

The MVR feature only runs on a L2 forwarding mode that limits the group membership view to a port basis only. The receiver port is locally a member of the IPMVLAN and the IGMP snooping function directly classifies a group member into the IPMVLAN. Traffic is then forwarded from the source port and the receiver port that are members of the multicast group.

MVR supports a L3 multicast forwarding mode that would allow a multicast group to be "routed" from the source port on the IPMVLAN to the receiver ports and receiver vlans that are member of the multicast group.

MVR eliminates the need to duplicate multicast traffic for subscribers in each VLAN. Multicast traffic is only sent around the VLAN trunk once—only on the multicast VLAN. The IGMP leave and join messages are in the VLAN to which the subscriber port is assigned. These messages dynamically register for streams of multicast traffic in the multicast VLAN on the Layer 3 device.

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 tag is the customer tag (c-tag).

Note. No checks is performed on c-tags as they are simple translation rules. VLAN addition or deletion rules do not affect them.

The following limitations has to 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

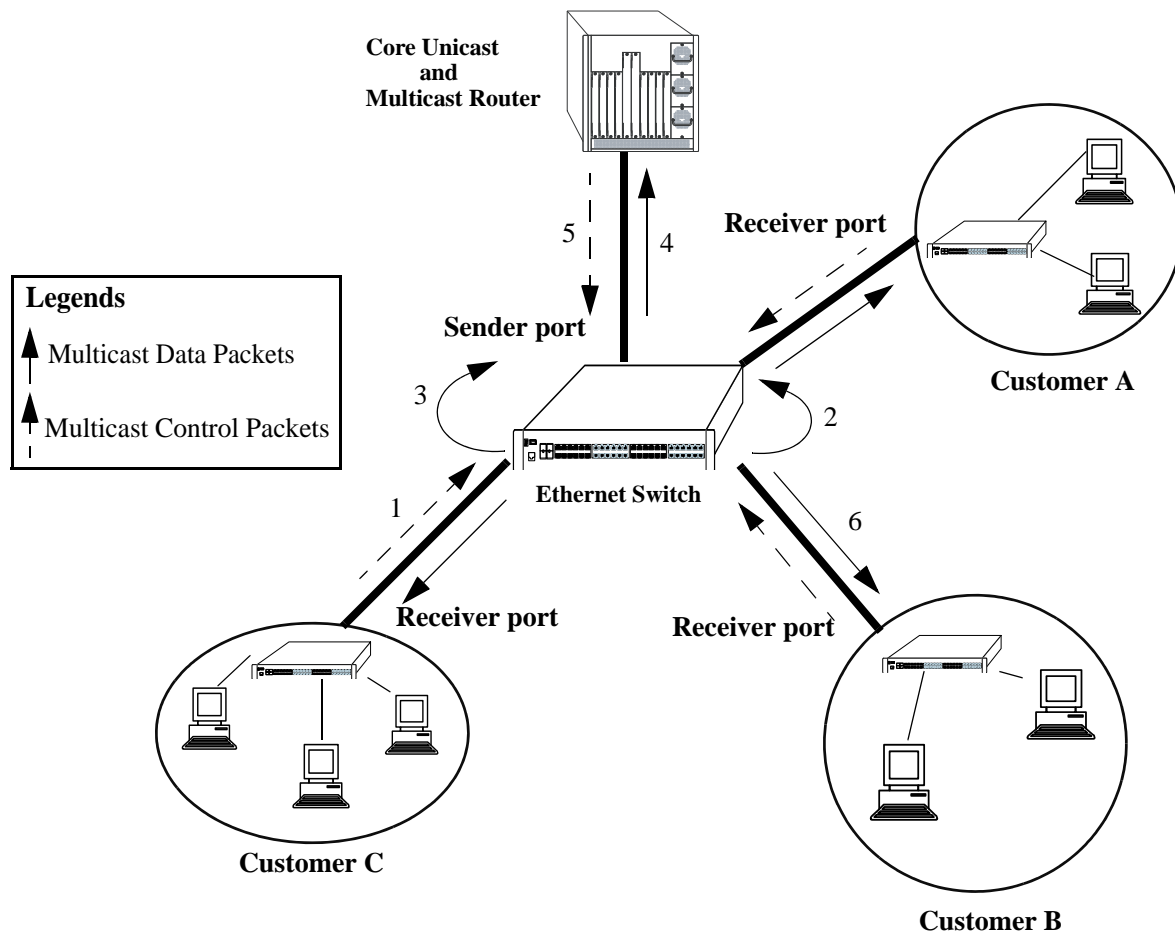
IP Multicast VLANs in the Enterprise mode contain normal user ports (fixed/tagged) as their member ports.

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 has to 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 has to 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 42-5](#)) 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 42-9](#)).
- Assigning IPv4/IPv6 address to an existing IPMVLAN and removing it (see [“Assigning and Deleting IPv4/IPv6 Address” on page 42-10](#)).
- Assigning and removing the c-tag in an IPMVLAN (see [“Assigning and Deleting a Customer VLAN Tag” on page 42-10](#)).
- Creating and deleting a sender port in an IPMVLAN (see [“Creating and Deleting a Sender Port” on page 42-10](#)).
- Creating and deleting a receiver port in an IPMVLAN (see [“Creating and Deleting a Receiver Port” on page 42-11](#)).
- Configuring a VLAN translation of a CVLAN to an IPMVLAN (see [“Associating an IPMVLAN with a Customer VLAN” on page 42-12](#)).

In addition, a tutorial is provided in [“IPMVLAN Application Example” on page 42-13](#) that shows you how to use CLI commands to configure a sample network.

Note. See the “IP Multicast VLAN Commands” chapter in the *OmniSwitch AOS Release 6 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 in the following example.

```
-> 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 in the following example.

```
-> 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 in the following example.

```
-> 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 in the following example.

```
-> 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 in the following example.

```
-> 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 in the following example.

```
-> 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 in the following example.

```
-> 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 has to 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 in the following example.

```
-> 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 in the following example.

```
-> 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 in the following example.

```
-> vlan ipmvlan 1003 receiver-port port 1/51
```

To configure the the port (or a range of ports) as receiver port for the IPMVLAN and associate RVLAN to receiver port (or a range of receiver ports), use the **vlan ipmvlan receiver-port** as shown in the following example.

```
-> vlan ipmvlan 1003 receiver-port port 1/51 receiver-vlan 10
```

In the VLAN Stacking mode, the port you want to configure as a receiver port has to 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 in the following example.

```
-> 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 in the following example.

```
-> no vlan ipmvlan 1003 receiver-port port 1/51
```

To delete receiver port association with a receiver VLAN, 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 in the following example.

```
-> no vlan ipmvlan 1003 receiver-port port 1/51 receiver-vlan 10
```

To delete Receiver port from an IPMVLAN, and delete all the RVLAN associations to the receiver port, 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 in the following example.

```
-> no vlan ipmvlan 1000 receiver-port port 1/51
```

The above CLI command will delete all the RVLAN associations to receiver 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 has to be a receiver port. Also, the receiver port must be a VLAN Stacking user port (UNI). For example, the following series of commands associates an IPMVLAN with a customer VLAN:

```
-> vlan ipmvlan 1002  
-> vlan ipmvlan 1002 receiver-port port 1/1 receiver-vlan 10
```

This above command will associate the IPMVLAN to the receiver-port/receiver-vlan pair.

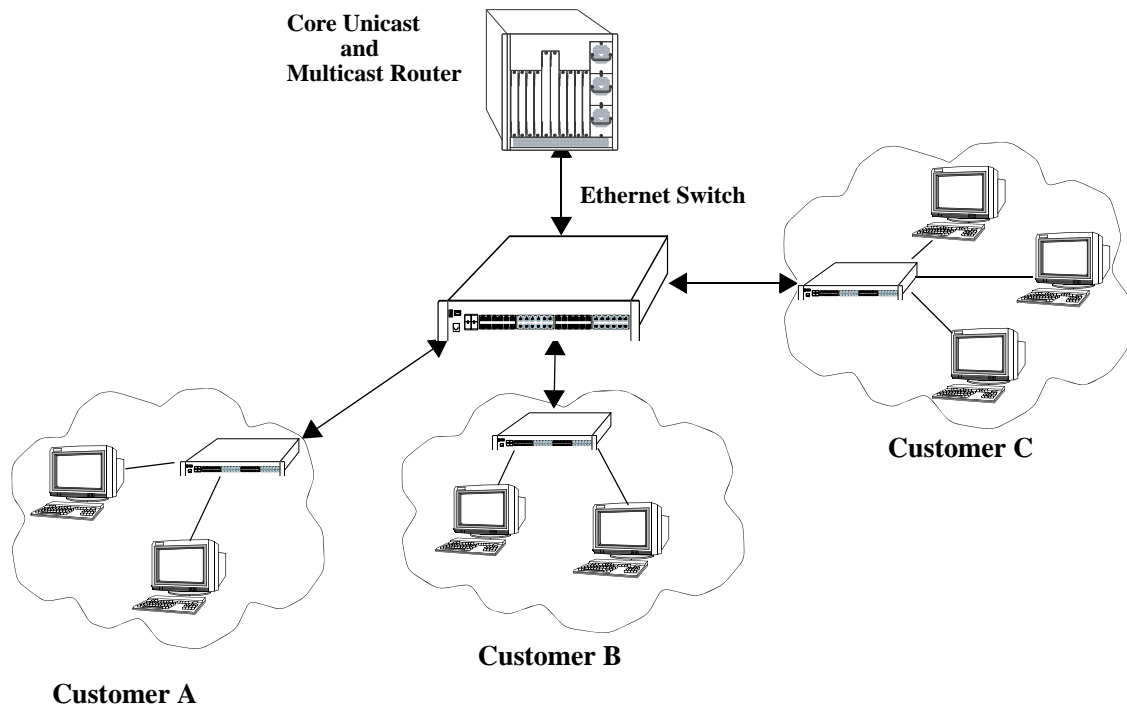
To associate an IPMVLAN with a linkagg, use the command as follows:

```
-> vlan ipmvlan 1002 receiver-port linkagg 1
```

Note. The maximum number of RVLANS that can be configured per receiver-port is limited to eight VLANs.

IPMVLAN Application Example

The figure in the following example 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

Perform the following steps 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 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 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
```

An example of what these commands look like when entered sequentially in E-service mode on the command line:

```
-> ethernet-service svlan 1000
-> ethernet-service ipmvlan 1001
-> ethernet-service svlan 1000 nni 1/23
-> ethernet-service service-name "customerA" svlan 1000 ethernet-service sap 10
service-name "customerA"
-> ethernet-service sap 10 uni 1/3
-> ethernet-service sap 10 cvlan 2000

-> vlan ipmvlan 1001 sender-port port 1/23
-> vlan ipmvlan 1001 receiver-port port 1/3 receiver-vlan 2000 vlan ipmvlan 1001
address 225.0.0.1
```

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. |

43 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 consumable resources, on the switch, 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 43-18 or [“Deleting A Mirroring Session”](#) on page 43-22.
- Protection from Spanning Tree changes (Port Mirroring)—see [“Unblocking Ports \(Protection from Spanning Tree\)”](#) on page 43-19.
- Enabling or Disabling Port Mirroring Status—see [“Enabling or Disabling Mirroring Status”](#) on page 43-20 or [“Disabling a Mirroring Session \(Disabling Mirroring Status\)”](#) on page 43-20.
- Configuring Port Mirroring Direction—see [“Configuring Port Mirroring Direction”](#) on page 43-20.
- Enabling or Disabling a Port Mirroring Session—see [“Enabling or Disabling a Port Mirroring Session \(Shorthand\)”](#) on page 43-21.

Port Monitoring

- Configuring a Port Monitoring Session—see [“Configuring a Port Monitoring Session”](#) on page 43-26.
- Enabling a Port Monitoring Session—see [“Enabling a Port Monitoring Session”](#) on page 43-26.
- Disabling a Port Monitoring Session—see [“Disabling a Port Monitoring Session”](#) on page 43-26.
- Deleting a Port Monitoring Session—see [“Deleting a Port Monitoring Session”](#) on page 43-26.
- Pausing a Port Monitoring Session—see [“Pausing a Port Monitoring Session”](#) on page 43-27.
- Configuring the persistence of a Port Monitoring Session—see [“Configuring Port Monitoring Session Persistence”](#) on page 43-27.
- Configuring a Port Monitoring data file—see [“Configuring a Port Monitoring Data File”](#) on page 43-27.
- Suppressing creation of a Port Monitoring data file—see [“Suppressing Port Monitoring File Creation”](#) on page 43-28.
- Configuring a Port Monitoring direction—see [“Configuring Port Monitoring Direction”](#) on page 43-28.
- Displaying Port Monitoring Status and Data—see [“Displaying Port Monitoring Status and Data”](#) on page 43-29.

sFlow

- Configuring a sFlow Session—see [“Configuring a sFlow Session”](#) on page 43-31.
- Configuring a Fixed Primary Address—see [“Configuring a Fixed Primary Address”](#) on page 43-32.
- Displaying a sFlow Receiver—see [“Displaying a sFlow Receiver”](#) on page 43-32.
- Displaying a sFlow Sampler—see [“Displaying a sFlow Sampler”](#) on page 43-33.
- Displaying a sFlow Poller—see [“Displaying a sFlow Poller”](#) on page 43-33.
- Displaying a sFlow Agent—see [“Displaying a sFlow Agent”](#) on page 43-33.
- Deleting a sFlow Session—see [“Deleting a sFlow Session”](#) on page 43-34.

RMON

- Enabling or Disabling RMON Probes—see [“Enabling or Disabling RMON Probes”](#) on page 43-36.

Switch Health Monitoring

- Configuring Resource Threshold Limits (Switch Health)—see [“Configuring Resource and Temperature Thresholds”](#) on page 43-42.
- Configuring Health Threshold Monitoring on Ports—see [“Enabling and Disabling Per-Port Health Threshold Monitoring”](#) on page 43-44
- Configuring Sampling Intervals—see [“Configuring Sampling Intervals”](#) on page 43-45.
- Resetting Health Statistics—see [“Resetting Health Statistics for the Switch”](#) on page 43-48.

For information about additional Diagnostics features such as Switch Logging and System Debugging/Memory Management commands, see [Chapter 44, “Using Switch Logging.”](#)

Note. The console messages "+++ healthMonCpuStatus Crossed Below The Threshold Limit " can be seen on switch bootup if it is configured to receive health monitoring debug messages on console or swlog file using the swlog appid and swlog output commands.

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 43-15](#).

Port Mirroring Specifications

| | |
|------------------------------|---|
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 128 |
| 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/ports, 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 43-22](#) or the [“Port Mirroring and Monitoring Commands” chapter in the *OmniSwitch AOS Release 6 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 43-25.

Port Monitoring Specifications

| | |
|-------------------------------|---|
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 43-25](#) or the “Port Mirroring and Monitoring Commands” chapter in the *OmniSwitch AOS Release 6 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 43-30](#).

sFlow Specifications

| | |
|---------------------|---|
| RFCs Supported | 3176 - sFlow Management Information Base |
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 43-30 or the “sFlow Commands” chapter in the *OmniSwitch AOS Release 6 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 43-30 or the “sFlow Commands” chapter in the *OmniSwitch AOS Release 6 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 43-30 or the “sFlow Commands” chapter in the *OmniSwitch AOS Release 6 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 43-35](#).

RMON Specifications

| | |
|------------------------------------|---|
| RFCs Supported | 2819 - Remote Network Monitoring Management Information Base |
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 43-37](#), [“Displaying Statistics for a Particular RMON Probe” on page 43-38](#), or the [“RMON Commands”](#) chapter in the *OmniSwitch AOS Release 6 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 43-41](#).

Switch Health Specifications

| | |
|--|--|
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 (for example, 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 |
| Health Threshold Monitoring Status per port | health threshold port-trap | Enabled on all chassis ports. |

Quick Steps for Configuring Switch Health Threshold Limits

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 (for example, **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 43-45](#) or the [“Health Monitoring Commands” chapter in the *OmniSwitch AOS Release 6 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 (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?

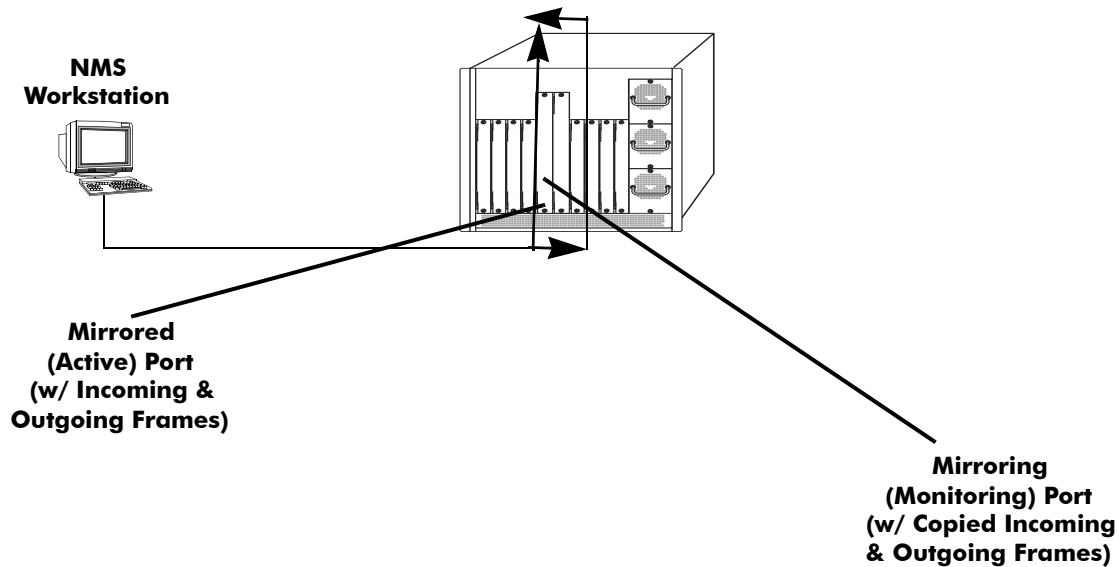
Mirroring between any 10/100/1000 port to any other 10/100/1000 port and between any SFP to any other SFP port is supported.

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

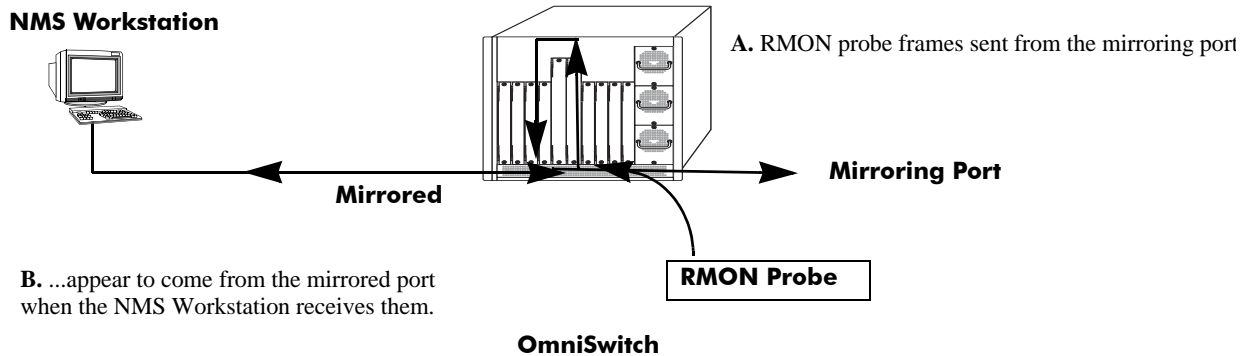
If mirroring is enabled on multiple ports and the same traffic is passing through these ports, then only one copy of each packet is sent to the mirroring destination. When the packet is mirrored for the first time, the switching ASIC flags the packet as “already mirrored.” If the packet goes through 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 (sent to CPU), whichever comes first.

Using Port Mirroring with External RMON Probes

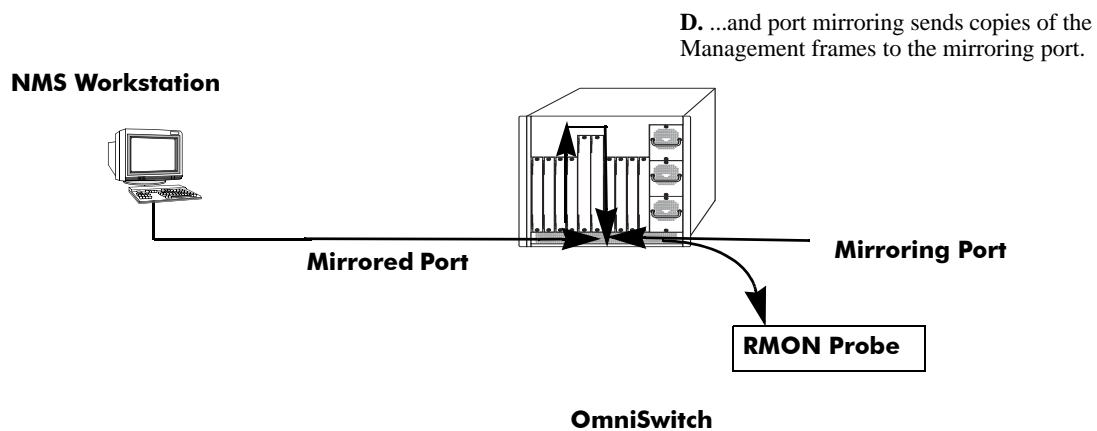
Port mirroring is a helpful monitoring tool when used in conjunction with an external RMON probe. Once you set up port mirroring, the probe can collect all relevant RMON statistics for traffic on the mirrored port. You can also move the mirrored port so that the mirroring port receives data from different 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 43-19 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 43-17](#).

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 7, “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 (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 (for example, source and destination slot/ports and optional unblocked VLAN ID number). See “[Enabling or Disabling a Port Mirroring Session \(Shorthand\)](#)” on [page 43-21](#) 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 (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
```

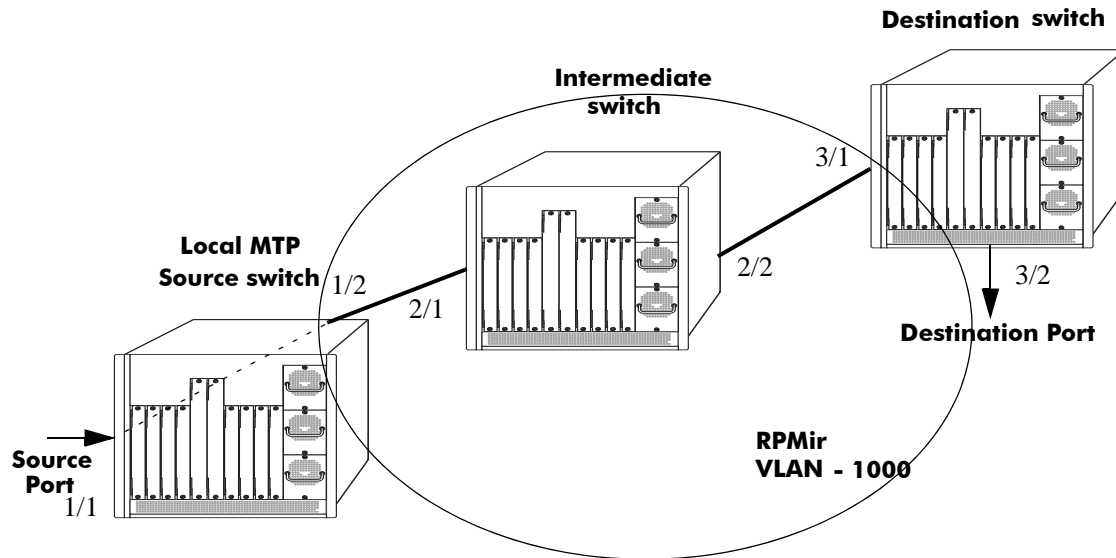
In this example, port mirroring session 6 is deleted.

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 (for example, 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 (sent to CPU), whichever comes first. See [“Mirroring on Multiple Ports” on page 43-16](#) 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 AOS Release 6 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 AOS Release 6 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 AOS Release 6 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 AOS Release 6 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; 6.1.1
Agent IP      = 127.0.0.1
```

Note. For more information about the displays that result from these commands, see the *OmniSwitch AOS Release 6 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
```

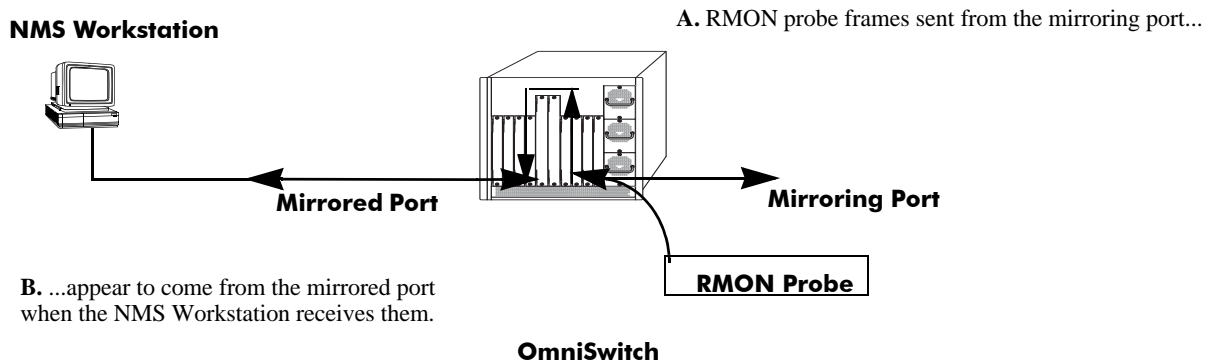
To delete a sFlow poller session, use the no form of the **sflow poller** command by entering **no sflow poller**, followed by the instance ID number, the slot number of the port to delete, a slash (/), and the port number of the port, enter:

```
-> no sflow poller 3 1/1
```

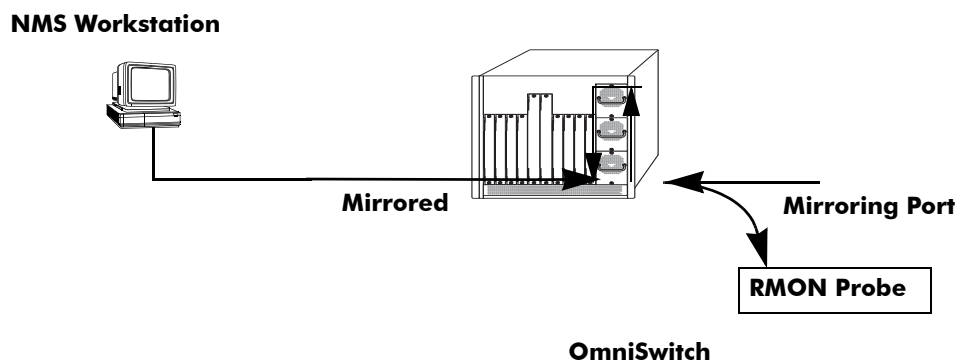
Remote Monitoring (RMON)

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 through 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
```

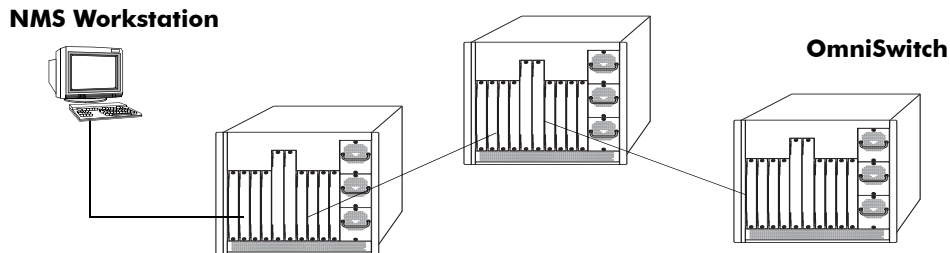
A display showing the specific logged RMON Event should appear, as shown in the following example:

| 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 43-42](#) for more information.
- **show health threshold**—Displays current health threshold settings. See [page 43-45](#) for details.
- **health threshold port-trap**—Enables or disables health threshold monitoring on a slot, port, or a range of ports. See [page 43-45](#) for details.
- **show health threshold port-trap**—Displays the current status of the health threshold settings for a slot, port, or a range of ports. See [page 43-45](#) for details.
- **health interval**—Configures sampling interval between health statistics checks. See [page 43-45](#) for more information.
- **show health interval**—Displays current health sampling interval, measured in seconds. See [page 43-46](#) for details.
- **show health**—Displays health statistics for the switch, as percentages of total resource capacity. See [page 43-46](#) for more information.
- **health statistics reset**—Resets health statistics for the switch. See [page 43-48](#) 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.

The **health threshold port-trap** command is used to enable or disable the health threshold monitoring on the user ports. Health threshold monitoring traps can be enabled only on uplink ports or only on a set of configured ports. This is to prevent the NMS from being flooded with too many threshold monitoring messages in big networks and prevent overwriting of threshold monitoring traps.

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). |

For example, to specify a CPU usage threshold of 85 percent, enter the following command:

```
-> health threshold cpu 85
```

Note. Do not configure the port health threshold (Rx and TxRx) value close to the line rate (rate at which traffic is sent). For example, if the traffic is sent at 50 % line rate, then configure the health threshold value of about 80% and not about 60%.

For more information on the **health threshold** command, refer to [Chapter 42, “Health Monitoring Commands,”](#) in the *OmniSwitch AOS Release 6 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.

Enabling and Disabling Per-Port Health Threshold Monitoring

Health threshold monitoring is enabled by default on all chassis ports. This command can be used to enable or disable health threshold monitoring on a slot, port, or a range of ports.

For example, to enable health threshold monitoring on slot 1, enter the following command:

```
-> health threshold port-trap 1 enable
```

To disable health threshold monitoring on port 1/2, enter the following command:

```
-> health threshold port-trap 1/2 disable
```

To disable health threshold monitoring on range of ports from 1/1 to 1/4, enter the following command:

```
-> health threshold port-trap 1/3-6 disable
```

To disable health threshold monitoring on slot 2, enter the following command:

```
-> health threshold port-trap 2 disable
```

Note. To verify health threshold monitoring settings for a slot, port, or a range of ports, use the **show health threshold port-trap** command. For example:

```
-> show health threshold port-trap 1
```

```
Slot/Port  Status
-----+-----
    1/1    enabled
    1/2    disabled
    1/3    disabled
    1/4    disabled
    1/5    disabled
    1/6    disabled
    1/7    enabled
    .
    .
    .
    1/26   enabled
```


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 43-42](#), as well as [Chapter 42, “Health Monitoring Commands,”](#) in the *OmniSwitch AOS Release 6 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
```

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
-----+-----+-----+-----+-----+-----
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
```

44 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 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 AOS Release 6 CLI Reference Guide*.

In This Chapter

The following procedures are described:

- [“Enabling Switch Logging” on page 44-6](#)
- [“Setting the Switch Logging Severity Level” on page 44-6](#)
- [“Specifying the Switch Logging Output Device” on page 44-9](#)
- [“Displaying Switch Logging Status” on page 44-10](#)
- [“Configuring the Switch Logging File Size” on page 44-10](#)
- [“Displaying Switch Logging Records” on page 44-11](#)

Note.

Switch logging commands are not intended for use with low-level hardware and software debugging. It is recommended that you contact an Alcatel Customer Service representative for assistance with debugging functions.

The console messages "+++ healthMonCpuStatus Crossed Below The Threshold Limit " can be seen on switch bootup if it is configured to receive health monitoring debug messages on console or swlog file using the swlog appid and swlog output commands.

Switch Logging Specifications

| | |
|---------------------------------|---|
| Platforms Supported | OmniSwitch 6350, 6450 |
| 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 file-size | 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 is sent.

```
-> swlog output console
```

In this example, the switch logging information is sent to the console port.

Note. *Optional.* To verify the switch logging configuration, enter the **show swlog** command. The display is similar to the following output:

```
-> show swlog
```

```
Operational Status           : On,
Log Device 1                  : flash,
Log Device 2                  : console,
Log Device 3                  : ipaddr 1.1.1.1
Syslog FacilityID            : local0(16),
Remote command-log           : Disabled,
Console Display Level         : debug3 (9),
All Applications Trace Level  : info (6)
```

```
Application ID      Level
-----+-----
BRIDGE              ( 10)  warning (5)
```

For more information about this command, or the “Switch Logging Commands” chapter in the *OmniSwitch AOS Release 6 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 is printed in the switch 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 AOS Release 6 Switch Management Guide*.

Switch Logging Commands Overview

This section describes the switch logging CLI commands, for enabling or disabling switch logging, displaying the 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 appears 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 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 are 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 is 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 44-8](#).

| 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 |

Note. The console messages "+++ healthMonCpuStatus Crossed Below The Threshold Limit " can be seen on switch bootup if it is configured to receive health monitoring debug messages on console or swlog file using the swlog appid and swlog output commands.

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 44-6](#). The severity level can be expressed by using either the severity level number or the severity level type as shown in the following table:

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 must 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. |

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 appears on the screen.

Specifying the Switch Logging Output Device

The **swlog output flash file-size** 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 or addresses.

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 appears 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 command:

```
-> swlog output flash
```

To disable the switch logging output to flash memory, enter the following command:

```
-> no swlog output flash
```

No confirmation message appears on the screen for either command.

Specifying an IP Address for Switch Logging Output

To specify a particular IP address destination (for example, a server) for switch logging output, enter the **swlog output flash file-size socket ipaddr** command, specifying the target IP address to which output is 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 appears on the screen.

Note. You can also send syslog files to multiple hosts (maximum of twelve).

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 appears on the screen.

To disable a specific configured output IP address from receiving switch logging output, command as specify an IPv4 or IPv6 address along with the **no swlog output** command. 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

Operational Status           : On,
Log Device 1                 : flash,
Log Device 2                 : console,
Syslog FacilityID           : system(3),
Remote command-log          : Disabled,
Console Display Level        : info (6),
All Applications Trace Level : info (6)
```

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 AOS Release 6 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 causes the switch to clear all the switch logging information and begin recording again. As a result, the switch displays a shorter file when you execute the [show log swlog](#) command. You can use the [swlog clear](#) command when the switch logging display is too long due to some of the data being old or out of date.

No confirmation message appears 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 AOS Release 6 CLI Reference Guide*. The following sample screen output shows a display of all the switch logging information.

Note. Switch logging frequently records a 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 successful!
MON NOV 11 13:18:24 2005    WEB       info The HTTP session login successful!
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 successful!
```

The fields in the [show log swlog](#) output are defined as follows:

- The **FILE ID** field specifies the File name (for example, swlog1.log), endPtr Global Sequence ID reference number (for example, 9968), Configuration Size (for example, 10000), Current Size (for example, 10000), and Mode (for example, 2).
- The **Timestamp** field indicates when the swlog entry occurred (for example, MON, NOV 11, 12:42:11 2005).
- The **Application** field specifies the application ID for which the stored swlog information is displayed (for example, SYSTEM).

- The **Level** field specifies the severity level for which the stored information is displayed (for example, 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.

When the switch is in ASA enhanced mode, both user name and password is prompted to view the SWLOG data using **show log swlog** command. Only those users who provide valid ASA credentials are allowed to view the data. For more information on Authenticated Switch Access - Enhanced Mode mode, refer chapter Managing Switch Security in *OmniSwitch AOS Release 6 Switch Management Guide*.

For example,

```
-> show log swlog
Username: test
Password:  *****
```


A Software License and Copyright Statements

This appendix contains Alcatel and third-party software vendor license and copyright statements.

Alcatel License Agreement

ALE USA, Inc. 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 Alcatel 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 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. **ALE USA, Inc.’s Rights.** Licensee acknowledges and agrees that the Licensed Materials are the sole property of Alcatel 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 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 considers the Licensed Files to contain valuable trade secrets of Alcatel, the unauthorized disclosure of which could cause irreparable harm to Alcatel. 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 harmless from any claim, lawsuit, legal proceeding, settlement or judgment (including without limitation Alcatel'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 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 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 for either replacement or, if so elected by Alcatel, 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 ALE USA, Inc. 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'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 for the Licensed Materials. IN NO EVENT SHALL ALE USA, Inc. BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL, OR EXEMPLARY DAMAGES OR LOST PROFITS, EVEN IF ALE USA, Inc. 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 and Licensee, if any, Alcatel is under no obligation to maintain or support the copies of the Licensed Files made and distributed hereunder and Alcatel 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 and certifying to Alcatel 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 may terminate this License Agreement upon the breach by Licensee of any term hereof. Upon such termination by Alcatel, Licensee agrees to return to ALE USA, Inc. ALE USA, Inc. 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 ALE USA, Inc.'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 ALE USA, Inc. 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 part 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 ALE USA, Inc. for a limited period of time. ALE USA, Inc. 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 “copy-right” 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 ALE USA, Inc.. 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 ALE USA, Inc. certain warranties of performance, which warranties [or portion thereof] ALE USA, Inc. 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 ALE USA, Inc. and Licensee, Licensee shall immediately return the EMWEB Product and any back-up copy to ALE USA, Inc., and will certify to ALE USA, Inc. 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 ALE USA, Inc. 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 acknowledgment: 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 39-19
qos port servicing mode command 39-27
qos stats interval command 39-22

Numerics

10/100/1000 ports
 defaults 1-4
802.1AB 22-1
 defaults 22-2
 specifications 22-2
 verify information about 22-24
802.1p
 trusted ports 39-32
802.1Q 24-1
 application examples 24-7
 defaults 24-2
 enabling notification 22-16
 enabling tagging 24-5
 frame type 24-6
 overview 24-3
 specifications 24-2
 trusted ports 39-6, 39-33
 verify information about 24-10
802.1Q ports
 trusted 39-32
802.1X 35-15, 37-1
 accounting 37-8
 and DHCP 37-7
 components 37-6
 defaults 37-3
 port authorization 37-10
 port parameters 35-30, 37-10
 port timeouts 37-10
 re-authentication 37-8, 37-12
 specifications 35-4, 37-2
802.1x command 37-3
802.1x initialize command 37-12
802.1x re-authenticate command 37-12
802.3ad
 see dynamic link aggregation

A

aaa accounting 802.1x command 37-12
aaa ace-server clear command 36-9
aaa authentication 802.1x command 37-9
 and 802.1X port behavior 37-7
aaa ldap-server command
 LDAP authentication 36-45
aaa radius-server command 37-9

RADIUS authentication 36-25, 36-32
Access Control Lists
 see ACLs
access list 30-15
 creating 30-15
Access Loop 16-5
Access Node 16-5
Access Node Identifier 16-7
accounting 1-34
ACLs
 application examples 40-4, 40-22
 bridged traffic 40-6
 defaults 40-3
 disposition 40-5, 40-7
 interaction with VRRP 34-10, 34-20
 Layer 2 40-11
 Layer 2 application examples 40-11
 Layer 3 40-12
 Layer 3 application examples 40-12
 multicast 40-14
 security features 40-15
 verify information about 40-20
actions
 combined with conditions 39-9, 39-11
 creating policy actions 39-38
 for ACLs 40-10
Address Resolution Protocol
 see ARP
advertisements 31-6
 destination address 31-9
 IP address preference 31-10
 lifetime 31-10
 transmission interval 31-9
Alcatel Mapping Adjacency Protocol 23-1
alerts 44-8
AMAP
 see Alcatel Mapping Adjacency Protocol
amap common time command 23-6
amap disable command 23-5
amap discovery time command 23-5
amap enable command 23-5
Application example
 Learned Port Security Configuration 3-3
application example
 Ethernet OAM 18-3
 MST 11-13
 MSTI 11-15
application examples
 802.1Q 24-7
 ACLs 40-4
 assigning ports to VLANs 7-3
 authentication servers 36-5
 Configuring 802.1AB 22-4
 DHCP Relay 32-7, 32-8, 32-11, 32-12
 dynamic link aggregation 26-4, 26-29
 GVRP 5-5
 ICMP policies 39-74
 interswitch protocols 23-8
 IP 28-4

- IPMS 41-40, 41-42
 - IPv6 29-3
 - Layer 2 ACLs 40-11
 - Layer 3 ACLs 40-12
 - mobile ports 7-3, 7-6, 7-8
 - policies 39-67
 - policy map groups 39-61
 - Port Mapping 8-2, 8-6
 - port mirroring 43-5
 - port monitoring 43-6, 43-9
 - QoS 39-35, 39-67
 - RDP 31-3
 - RIP 30-3
 - RMON 43-12
 - Server Load Balancing 33-3
 - source learning 2-3
 - Spanning Tree Algorithm and Protocol 12-13, 12-44
 - static link aggregation 25-3, 25-11
 - switch health 43-14
 - switch logging 44-4
 - UDLD 20-3
 - VLAN advertisements 5-4
 - VLAN rules 9-3, 9-14
 - VLANs 4-3, 4-10, 7-3
 - VRRP 34-5, 34-28, 34-32
 - VRRP3 34-33
 - applied configuration 39-64
 - how to verify 39-66
 - ARP
 - clearing the ARP cache 28-14
 - creating a permanent entry 28-13
 - deleting a permanent entry 28-14
 - dynamic entry 28-13
 - filtering 28-17
 - local proxy 28-14
 - arp command 28-13
 - arp filter command 28-17
 - assigning ports 4-6
 - assigning ports to VLANs 7-1
 - application examples 7-3
 - defaults 7-3
 - dynamic port assignment 7-5
 - static port assignment 7-4
 - Authenticated Switch Access
 - LDAP VSAs 36-41
 - authentication servers
 - application example 36-5
 - defaults 36-3
 - how backups work 36-7
 - see* LDAP authentication servers, RADIUS authentication servers
 - automatic IP configuration 32-17
- B**
- backup router
 - VRRP 34-7
 - boundary port 11-11
 - BPDU
 - see* Bridge Protocol Data Units
 - bridge 1x1 forward delay command 12-27
 - bridge 1x1 hello time command 12-26
 - bridge 1x1 protocol command 12-24
 - bridge 1x1 slot/port command 12-33
 - bridge 1x1 slot/port admin-edge command 12-40
 - bridge 1x1 slot/port path cost command 12-36
 - bridge auto-vlan-containment command 12-29
 - bridge cist forward delay command 12-27
 - bridge cist hello time command 12-26
 - bridge cist protocol command 12-24
 - bridge cist slot/port admin-edge command 12-40
 - bridge forward delay command 12-27
 - bridge hello time command 12-26
 - bridge max age command 12-26
 - bridge mode command 12-15
 - bridge msti priority command 12-25
 - bridge path cost mode command 12-28
 - bridge priority command 12-25
 - bridge protocol command 12-24
 - Bridge Protocol Data Units
 - contents 12-10
 - bridge slot/port command 12-28
 - bridge slot/port connection command 12-39
 - bridge slot/port path cost command 12-36
 - bridge slot/port priority command 12-34
 - built-in port groups 39-14
 - used with Policy Based Routing 39-75
- C**
- Circuit Identifier 16-7
 - clear arp filter command 28-17
 - clear arp-cache command 28-14
 - Client 16-6
 - combo ports 1-4
 - configuring 1-18
 - overview 1-4
 - condition groups
 - for ACLs 39-50, 40-8
 - MAC groups 39-54, 39-58
 - network groups 39-51
 - port groups 39-55
 - sample configuration 39-50
 - service groups 39-53
 - verify information about 39-59
 - conditions
 - combined with actions 39-9, 39-11
 - configuring 39-37
 - for ACLs 40-9
 - how to create 39-37
 - see also* condition groups
 - testing before applying 39-48
 - valid combinations 39-7
 - valid combinations for ACLs 40-6
 - Configuring 802.1AB
 - application examples 22-4
 - counters 1-34

D

- debug messages 44-8
- debug qos command 39-18
- default route
 - IP 28-12
- defaults
 - 10/100/1000 ports 1-4
 - 802.1AB 22-2
 - 802.1Q 24-2
 - 802.1X 37-3
 - ACLs 40-3
 - assigning ports to VLANs 7-3
 - authentication servers 36-3
 - DHCP Relay 32-5, 32-6
 - DVMRP 5-2
 - dynamic link aggregation 26-3, 27-4
 - Ethernet OAM 17-2, 18-2
 - Ethernet ports 1-3, 1-4, 16-2
 - interswitch protocols 23-2
 - IP 28-4
 - IPMS 41-4, 41-5
 - IPv6 29-3
 - Learned Port Security 3-2
 - mobile ports 7-3
 - Multiple Spanning Tree 12-6
 - policy servers 38-2
 - Port Mapping 8-2
 - port mirroring 43-4
 - port monitoring 43-6, 43-8
 - QoS 39-12
 - RDP 31-2
 - RDP interface 31-8
 - RIP 30-2
 - RMON 43-12
 - RRSTP 12-7
 - source learning 2-2
 - Spanning Tree Bridge 12-4, 13-3
 - Spanning Tree Port 12-5
 - static link aggregation 25-2
 - switch health 43-14
 - switch logging 44-3
 - UDLD 20-2
 - VLAN rules 9-2
 - VLANs 4-3
 - VRRP 34-3
- Denial of Service
 - see* DoS
- DHCP 32-10
 - used with 802.1X 37-7
- DHCP Relay 32-1, 32-15, 32-45
 - application examples 32-7, 32-8, 32-11, 32-12
 - AVLAN forwarding option 32-16, 32-45
 - defaults 32-5, 32-6
 - DHCP server IP address 32-14, 32-44
 - forward delay time 32-16
 - maximum number of hops 32-16, 32-45
 - standard forwarding option 32-16, 32-45
- DHCP VLAN rules 9-5
- directed broadcast 28-26
- disposition 40-10
 - ACLs 40-5, 40-7
 - global defaults for QoS rules 39-16
- DoS 28-27
 - enabling traps 28-31
 - setting decay value 28-31
 - setting penalty values 28-30
 - Setting Port Scan Penalty Value 28-30
- DSCP
 - trusted ports 39-32
- DVMRP
 - defaults 5-2
- dynamic link aggregation 26-1, 27-1
 - application examples 26-4, 26-29
 - defaults 26-3, 27-4
 - group actor administrative key 26-15
 - group actor system ID 26-16
 - group actor system priority 26-15
 - group administrative state 26-14
 - group partner administrative key 26-16
 - group partner system ID 26-17
 - group partner system priority 26-17
 - groups 26-10
 - assigning ports 26-11
 - creating groups 26-10
 - deleting groups 26-10
 - group names 26-14
 - removing ports 26-12
 - LACPDU bit settings 26-18, 26-22
 - LACPDU frames 26-18, 26-22
 - Link Aggregation Control Protocol (LACP) 26-6
 - MAC address 26-16, 26-17, 26-20, 26-24
 - port actor administrative priority 26-20
 - port actor port priority 26-21
 - port actor system administrative states 26-18
 - port actor system ID 26-20
 - port partner administrative key 26-24
 - port partner administrative priority 26-26
 - port partner administrative state 26-22
 - port partner administrative system ID 26-24
 - port partner administrative system priority 26-25
 - port partner port administrative status 26-26
 - ports 26-11
 - specifications 26-2, 27-3
 - verify information about 26-33, 27-14
- dynamic log
 - LDAP accounting servers 36-44
- dynamic VLAN port assignment
 - mobile ports 7-5
 - secondary VLANs 7-13
 - VLAN rules 9-1

E

- error frame 1-34
- errors 44-8
- Ethernet
 - defaults 1-3, 1-4, 16-2

- flood rate 1-9
- frame size 1-12
- full duplex 1-14, 1-19
- half duplex 1-14, 1-19
- specifications 1-2
- verify information 1-34

Ethernet OAM

- application example 18-3
- configuration 17-10, 18-3
- Connectivity Fault Management
 - Continuity Check Messages 17-6
 - Link Trace Messages 17-6
 - Loop-back Messages 17-6
- defaults 17-2, 18-2
- overview 17-3
- specifications 17-2, 18-2
- verification 17-17

ethoam association ccm-interval command 17-11

ethoam association command 17-9

ethoam association mhf command 17-11, 17-12

ethoam association-default command 17-11

ethoam domain command 17-9

ethoam end-point command 17-9

ethoam intermediate-point command 17-9

ethoam linktrace command 17-13

ethoam loopback command 17-12

F

Fast Spanning Tree 12-8

filtering lists

- see* ACLs

flow command 1-17, 1-22

frame type 24-6

G

GARP

- active member 5-3
- messages 5-2
- 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 portcommand 5-5

gvrp registration command 5-9

gvrp static-vlan restrictcommand 5-5

GVRP Timers 5-10

gvrp transparent switchingcommand 5-8

gvrpcommand 5-5

H

health interval command 17-13, 43-45

health statistics reset command 43-48

health threshold command 43-42

health threshold limits

- displaying 43-45

Hot Standby Routing Protocol

- see* HSRP

HSRP

- not compatible with VRRP 34-3

I

ICMP 28-34

- control 28-37
- QoS policies for 39-74
- statistics 28-37

icmp messages command 28-36

icmp type command 28-35, 28-36

IEEE 24-1

IGMP

- multicast ACLs 40-1, 40-14

IGMP Spoofing 41-20

Institute of Electrical and Electronics Engineers

- see* IEEE

interfaces admin command 1-8

interfaces alias command 1-12

interfaces autoneg command 1-16

interfaces crossover command 1-16

interfaces duplex command 1-14, 1-24

interfaces hybrid autoneg command 1-20

interfaces hybrid crossover command 1-21

interfaces hybrid duplex command 1-19

interfaces hybrid speed command 1-18

interfaces ifg command 1-15

interfaces max frame command 1-12

interfaces no l2 statistics command 1-8

interfaces speed command 1-14

inter-frame gap 1-34, 1-35

inter-frame gap value 1-15

Intermediate Agent 16-1

Internet Control Message Protocol

- see* ICMP

interswitch protocols

- AMAP 23-1, 23-3
- application examples 23-8
- defaults 23-2
- specifications 23-2

IP 28-1

- application examples 28-4
- ARP 28-13
- defaults 28-4
- directed broadcast 28-26
- ICMP 28-34
- ping 28-37
- protocols 28-5
- router ID 28-18
- router port 28-7
- router primary address 28-18

- specifications 28-2
 - static route 28-10, 29-12
 - tracing an IP route 28-38
 - TTL value 28-19
 - UDP 28-38
 - verify information about 28-44
 - ip access-list address command 30-15
 - ip access-list command 30-15
 - ip default-ttl command 28-19
 - ip directed-broadcast command 28-26
 - ip dos scan close-port-penalty command 28-30
 - ip dos scan decay command 28-31
 - ip dos scan tcp open-port-penalty command 28-30
 - ip dos scan threshold command 28-30
 - ip dos scan udp open-port-penalty command 28-30
 - ip dos trap command 28-31
 - ip helper address command 32-14, 32-44
 - ip helper boot-up command 32-17
 - ip helper forward delay command 32-16
 - ip helper maximum hops command 32-16, 32-45
 - ip helper per-vlan command 32-16, 32-45
 - ip helper standard command 32-16, 32-45
 - ip interface command 30-3
 - ip load rip command 30-3, 30-7
 - ip multicast igmp-proxy-version command 41-10, 41-26
 - ip multicast neighbor-timeout command 41-10, 41-17, 41-18, 41-26, 41-33
 - ip multicast query-interval command 41-15, 41-16, 41-30
 - ip multicast static-member command 41-14
 - ip multicast static-neighbor command 41-27
 - ip multicast static-querier command 41-12
 - IP Multicast Switching
 - see IPMS
 - ip multicast switching command 41-8, 41-20, 41-25, 41-35
 - IP multinetting 28-6
 - ip redist command 30-12
 - ip rip force-holddowntimer command 30-10
 - ip rip garbage-timer command 30-11
 - ip rip holddown-timer command 30-11
 - ip rip host-route command 30-11
 - ip rip interface auth-key command 30-18
 - ip rip interface auth-type command 30-18
 - ip rip interface command 30-3, 30-7
 - ip rip interface metric command 30-9
 - ip rip interface recv-version command 30-8
 - ip rip interface send-version command 30-8
 - ip rip interface status command 30-3, 30-8
 - ip rip invalid-timer command 30-10
 - ip rip route-tag command 30-9
 - ip rip status command 30-3, 30-7
 - ip rip update-interval command 30-10
 - ip route-pref command 28-18
 - IP router ports 28-7
 - modifying 28-9
 - removing 28-9
 - ip router primary-address command 28-18
 - ip router router-id command 28-18
 - ip router-discovery command 31-3, 31-8
 - ip router-discovery interface advertisement-address
 - command 31-9
 - ip router-discovery interface advertisement-lifetime
 - command 31-10
 - ip router-discovery interface max-advertisement-interval
 - command 31-9
 - ip router-discovery interface min-advertisement-interval
 - command 31-10
 - ip router-discovery interface preference-level
 - command 31-10
 - ip service command 28-32
 - ip slb admin command 42-9
 - ip static-route command 28-10, 29-12
 - IPMS 41-1
 - adding static members 41-14, 41-38, 41-39
 - adding static neighbors 41-11
 - adding static queriers 41-12
 - application examples 41-40, 41-42
 - defaults 41-4, 41-5
 - deleting static members 41-14, 41-29
 - deleting static neighbors 41-12
 - deleting static queriers 41-13, 41-28
 - displaying 41-44, 41-45
 - enabling 41-8, 41-20, 41-21, 41-22, 41-35, 41-36, 41-37
 - IGMPv2 41-11, 41-27
 - IGMPv3 41-10, 41-26
 - neighbor timeout 41-17, 41-18, 41-19, 41-32, 41-34
 - overview 41-6
 - query interval 41-15, 41-16, 41-30, 41-31
 - RFCs 41-3
 - specifications 41-3
 - IPMV
 - ipv4, ipv6 address 42-16
 - IPv6 29-1
 - addressing 29-5
 - application examples 29-3
 - autoconfiguration of addresses 29-7
 - defaults 29-3
 - specification 29-2
 - verify information about 29-21
 - ipv6 access-list address command 30-15
 - ipv6 access-list command 30-15
 - ipv6 address command 29-3, 29-11
 - ipv6 interface command 29-3, 29-9
 - ipv6 load rip command 29-4
 - ipv6 rip interface command 29-4
 - ipv6 route-pref command 29-13
- J**
- jumbo frames 1-2
- L**
- LACP
 - see dynamic link aggregation
 - lacp agg actor admin key command 26-4, 26-11
 - lacp agg actor admin state command 26-18
 - lacp agg actor port priority command 26-21
 - lacp agg actor system id command 26-20
 - lacp agg actor system priority command 26-20

- lACP agg partner admin key command 26-24
 - lACP agg partner admin port command 26-26
 - lACP agg partner admin port priority command 26-26
 - lACP agg partner admin state command 26-22
 - lACP agg partner admin system id command 26-24
 - lACP agg partner admin system priority command 26-25
 - lACP linkagg actor admin key command 26-15
 - lACP linkagg actor system id command 26-16
 - lACP linkagg actor system priority command 26-15
 - lACP linkagg admin state command 26-14
 - lACP linkagg name command 26-14
 - lACP linkagg partner admin key command 26-16
 - lACP linkagg partner system id command 26-17
 - lACP linkagg partner system priority command 26-17
 - lACP linkagg size command 26-4, 26-10
 - Layer 2
 - statistics counters 1-8
 - LDAP accounting servers
 - dynamic log 36-44
 - standard attributes 36-42
 - LDAP authentication servers
 - directory entries 36-36
 - functional privileges 36-41
 - passwords for 36-40
 - schema extensions 36-36
 - SNMP attributes on authentication servers 36-42
 - SSL 36-46
 - VSAs for Authenticated Switch Access 36-41
 - LDAP servers
 - see* policy servers
 - used for QoS policies 38-3
 - Learned Port Security
 - database table 3-8
 - defaults 3-2
 - disabling 3-9
 - enabling 3-9
 - overview 3-5
 - specifications 3-2
 - Learned Port Security Configuration
 - Application example 3-3
 - Lightweight Directory Access Protocol
 - see* LDAP servers
 - line speed 1-14, 1-18
 - link aggregation
 - 802.1Q 24-5
 - dynamic link aggregation 26-1, 27-1
 - enabling tagging 24-5
 - Spanning Tree parameters 12-33, 12-35, 12-37, 12-38, 12-40
 - static link aggregation 25-1
 - lldp lldpdu command 22-4
 - lldp notification command 22-4
 - lldp tlv dot1 command 22-17
 - lldp tlv dot3 command 22-18
 - lldp tlv management command 22-4
 - lldp tlv med command 22-18, 22-19
 - logged events
 - detail level 39-19
 - sent to PolicyView 39-19
 - types of events 39-18
- 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 9-5
 - MAC addresses
 - aging time 2-9, 12-27
 - dynamic link aggregation 26-16, 26-17, 26-20, 26-24
 - learned 2-5
 - statically assigned 2-5
 - mac-address-table command 2-5
 - mac-address-table-aging-time command 2-9
 - map groups 39-61
 - application 39-74
 - creating 39-62
 - verifying information 39-63
 - master router
 - VRRP 34-7
 - MLD Zapping 41-36
 - mobile port properties 7-16
 - BPDU ignore 7-11
 - default VLAN membership 7-12
 - restore default VLAN 7-12
 - mobile ports 7-11
 - application examples 7-3, 7-6, 7-8
 - defaults 7-3
 - dynamic VLAN port assignment 7-5, 7-12
 - secondary VLANs 7-13
 - trusted 39-6, 39-32
 - VLAN rules 9-1
 - MST 11-4
 - application example 11-13
 - Internal Spanning Tree (IST) Instance 11-9
 - Interoperability 11-11
 - Migration 11-11, 11-12
 - MSTI 11-7
 - application example 11-15
 - MSTP 11-4
 - Multiple Spanning Tree Region 11-8
 - Multicast Listener Discovery (MLD) 41-26
 - Multiple Spanning Tree
 - defaults 12-6
- N**
- network address VLAN rules 9-5
 - non combo ports
 - configuring 1-14
- O**
- OSPF redistribution policies
 - deleting 28-22, 28-24, 29-15, 29-18, 30-16

P

- pending configuration 39-64
- pending policies
 - deleting 39-65
 - testing 39-48
- Per VLAN DHCP 32-15, 32-44
- ping
 - IP 28-37
- ping command 28-37
- policies
 - application examples 39-67
 - applied 39-64
 - built-in 39-14
 - conditions 39-37
 - creating policy actions 39-38
 - how the switch uses them 39-4
 - Policy Based Routing 39-75
 - precedence 39-41, 40-6
 - redirect linkagg 39-72
 - redirect port 39-72
 - rules 39-39
 - verify information about 39-47
- policies configured via PolicyView 39-66
- policy
 - for ACLs 40-10
 - policy actions 40-10
 - policy conditions 40-9
 - policy rule 40-10
- policy action 802.1p command 39-33
- policy action command 39-24, 39-34
- policy action map command 39-61
- policy action redirect linkagg command 39-72
- policy action redirect port command 39-72, 39-73
- policy actions
 - see* actions
- Policy Based Routing 39-75
- policy condition command 39-34
- policy conditions
 - see* conditions
- policy mac group command 39-50, 40-8
- policy MAC groups 39-54, 39-58
- policy map group command 39-61
- policy map groups
 - application example 39-61
- policy network group command 39-50, 40-8
- policy network groups 39-51
 - switch default group 39-14, 39-51
- policy port group command 39-50, 40-8
- policy port groups 39-55
- policy rule command 39-34
- policy server command 38-2, 38-4
- policy server flush command 38-7
 - compared to qos flush command 38-7
- policy server load command 38-6
- policy servers
 - defaults 38-2
 - downloading policies 38-6
 - installing 38-3
 - SSL 38-6
- policy service command 40-8
- policy service group command 39-50, 40-8
- policy service groups 39-53
- policy services 39-52
- PolicyView
 - LDAP policy servers 38-1
- Port Based Network Access Control
 - see* 802.1X
- Port Mapping 8-1
 - application examples 8-2, 8-6
 - defaults 8-2
 - specifications 8-2
- port mapping command 8-2
- Port Mapping Session
 - creating and deleting 8-3
 - enabling and disabling 8-4
- port mirroring 18-2, 43-15
 - application examples 43-5
 - defaults 43-4
 - direction 43-20
 - disabling mirroring status 43-20
 - displaying status 43-22
 - enabling or disabling mirroring status 43-20
 - N-to-1 port mirroring 43-19
 - specifications 43-4
 - unblocking ports 43-19
- port mirroring command 43-21
- port mirroring session
 - creating 43-18
 - deleting 43-22
 - enabling/disabling 43-21
- port mirroring source command 43-6
- port mirroring source destination command 43-18, 43-20
- port mobility
 - see* mobile ports
- port monitoring
 - application examples 43-6, 43-9
 - configuring 43-26, 43-31, 43-32
 - creating a data file 43-27
 - defaults 43-6, 43-8
 - deleting a session 43-26, 43-34
 - direction 43-28
 - disabling a session 43-26
 - displaying status and data 43-29, 43-32, 43-33
 - enabling a session 43-26
 - file overwriting 43-28
 - file size 43-27
 - overview 43-25, 43-30
 - pausing a session 43-27
 - resuming a session 43-27
 - session persistence 43-27
 - specifications 43-6, 43-8
 - suppressing file creation 43-28
- port monitoring command 43-26, 43-27
- port monitoring source command 43-26, 43-27, 43-28, 43-31
- port status 1-34
- port VLAN rules 9-6
- ports

- 802.1Q 24-5
 - displaying QoS information about 39-34
 - enabling tagging 24-5
 - mobile ports 7-11
 - Spanning Tree parameters 12-30
 - trusted 39-32
 - VLAN assignment 4-6, 7-1
- port-security command 3-9
- port-security shutdown command 3-10
- PPPoE Intermediate Agent 16-1
- Precedence
 - Configured rule order 39-41
 - Precedence value 39-41
- precedence
 - ACLs 40-6
 - Configured rule order 40-6
 - for policies 39-41, 40-6
 - Precedence value 40-6
- protocol VLAN rules 9-5

Q

QoS

- application examples 39-35, 39-67
- ASCII-file-only syntax 39-36
- configuration overview 39-15
- defaults 39-12
- enabled/disabled 39-15
- interaction with other features 39-6
- overview 39-3
- quick steps for creating policies 39-35
- Specifications 39-2
- traffic prioritization 39-68
- qos apply command 39-64
 - global configuration 39-64
 - policy and port configuration 39-64
 - testing conditions 39-48
- qos clear log command 39-22
- qos command 39-15
- qos default bridged disposition command 39-14, 39-16
- qos default bridged disposition command
 - for ACLs 40-7
- qos default multicast disposition command 39-14, 39-16
- qos default routed disposition command 39-14, 39-16
- qos default servicing mode command 39-16, 39-27
- qos flush command 39-65
 - compared to policy server flush command 38-7
- qos forward log command 39-19
- QoS log
 - cleared 39-22
 - displayed 39-21, 39-22
 - number of display lines 39-19
 - see also* logged events
- qos log level command 39-18, 39-19
- qos port command 39-24
- qos port default 802.1p command 39-32
- qos port default dscp command 39-32
- qos port q minbw maxbw command 39-28
- qos port trusted command 39-33

- qos reset command 39-23
- qos revert command 39-65
- qos stats interval command 39-22
- qos trust ports command 39-33
- qos user-port command 40-16
- Quality of Service
 - see* QoS
- queues
 - shared 39-24

R

- RADIUS accounting servers
 - standard attributes 36-14
 - used for 802.1X 37-12
 - VSAAs 36-15
- RADIUS authentication servers 36-10
 - functional privileges 36-14
 - standard attributes 36-10
 - used for 802.1X 37-6
 - VSAAs 35-71, 36-13
- Rapid Spanning Tree Algorithm and Protocol
 - see* RSTP
- RDP 31-1, 31-5
 - advertisement destination address 31-9
 - advertisement interval 31-9
 - advertisement lifetime 31-10
 - application examples 31-3
 - defaults 31-2
 - disable 31-8
 - enable 31-8
 - example 31-5
 - interface 31-6
 - IP address preference 31-10
 - security 31-7
 - specifications 31-2
 - verify information about 31-11
- RDP interface 31-6
 - defaults 31-8
- re-authentication
 - 802.1X 37-8
- Redirection Policies 39-72
- Remote Authentication Dial-In User Service
 - see* RADIUS authentication servers
- Remote Identifier 16-8
- resource threshold limits
 - configuring 43-42
- Ring Rapid Spanning Tree Algorithm and Protocol
 - see* RRSTP
- RIP 30-1
 - application examples 30-3
 - defaults 30-2
 - enabling 30-7
 - forced hold-down timer 30-10
 - garbage timer 30-11
 - hold-down timer 30-11
 - host route 30-11
 - interface 30-7
 - invalid timer 30-10

- IP 30-4
 - loading 30-7
 - redistribution 30-12
 - security 30-18
 - specifications 30-2
 - unloading 30-7
 - update interval 30-10
 - verification 30-19
 - verify information about 30-19
 - RIP interface
 - creating 30-7
 - deleting 30-7
 - enabling 30-8
 - metric 30-9
 - password 30-18
 - receive option 30-8
 - route tag 30-9
 - send option 30-8
 - RMON
 - application examples 43-12
 - defaults 43-12
 - specifications 43-11
 - RMON events
 - displaying list 43-39
 - displaying specific 43-40
 - RMON probes
 - displaying list 43-37
 - displaying statistics 43-38
 - enabling/disabling 43-36
 - rmon probes command 43-36
 - RMON tables
 - displaying 43-37
 - route map
 - creating 30-13
 - deleting 30-14
 - enabling/disabling administrative status 30-16
 - redistribution 30-16
 - sequencing 30-14
 - Router Discovery Protocol
 - see* RDP
 - router ID 28-18, 29-13
 - router port
 - IP 28-7
 - router primary address 28-18
 - Routing Information Protocol
 - see* RIP
 - RRSTP 12-42
 - configuration 12-43
 - defaults 12-7
 - RSTP 12-8
 - port connection types 12-39
 - rules
 - see* policies
- S**
- sampling intervals
 - configuring 17-13, 43-45
 - viewing 43-46
 - Secure Socket Layer
 - see* SSL
 - security 31-7
 - Security Violation Mode 3-20
 - restrict mode 3-20
 - Server Load Balancing
 - application examples 33-3
 - disabling 12-43
 - enabling 12-43
 - severity level
 - see* switch logging
 - shared queues 39-24
 - show 802.1q command 24-7, 24-10
 - show amap command 23-5, 23-7
 - show arp command 28-13
 - show arp filter command 28-17, 28-31
 - show bridge rrstp configuration command 12-43
 - show bridge rrstp ring command 12-43
 - show gvrp configuration port command 5-9
 - show health command 43-46
 - show health interval command 43-46
 - show health threshold command 43-14, 43-45
 - show icmp control command 28-37
 - show icmp statistics command 28-37
 - show ip config command 28-19, 28-26
 - show ip interface command 28-9
 - show ip redist command 30-16
 - show ip rip command 30-7
 - show ip rip interface command 30-7
 - show ip route command 28-11, 29-12
 - show ip route-map command 30-13
 - show ipv6 interface command 29-9
 - show linkagg command 25-12
 - show linkagg port command 25-12
 - show lldp remote-system command 22-4, 22-7
 - show lldp statistics command 22-4, 22-7
 - show log swlog command 44-11
 - show policy server long command 38-6
 - show port mirroring status command 43-22
 - show port monitoring file command 43-29
 - show port-security command 3-4
 - show port-security shutdown command 3-4
 - show qos log command 39-21, 39-22
 - show rmon events command 43-37
 - show rmon probes command 43-12, 43-37
 - show spantree command 12-15
 - show spantree mst regioncommand 11-15
 - show swlog command 44-4, 44-10
 - show tcp ports command 28-38
 - show tcp statistics command 28-38
 - show uddl configuration command 20-3
 - show uddl statistics port command 20-3
 - show udp ports command 28-38
 - show udp statistics command 28-38
 - show vlan svlan command 6-18, 10-37, 19-11
 - show vlan svlan port-binding command 6-18, 10-37, 19-11
 - show vlan svlan port-config command 6-18, 10-37, 19-11
 - SNMP
 - attributes for LDAP authentication servers 36-42

- source learning 2-1
 - application examples 2-3
 - defaults 2-2
 - MAC address table 2-1, 2-5
- source learning time limit 3-10
- Spanning Tree
 - specifications 12-3, 13-2
- Spanning Tree Algorithm and Protocol 12-1, 13-1
 - 1x1 operating mode 4-9, 12-15, 12-17
 - application examples 12-13, 12-44
 - bridge ID 12-10, 12-24
 - Bridge Protocol Data Units 7-11, 12-10, 12-25, 12-26, 12-27
 - bridged ports 12-30
 - designated bridge 12-8
 - flat operating mode 4-9, 12-15, 12-16
 - path cost 12-35
 - port connection types 12-39
 - Port ID 12-10
 - port ID 12-34
 - port path cost 12-8
 - port roles 12-8
 - port states 12-9, 12-38
 - root bridge 12-8, 12-25, 12-26, 12-27
 - root path cost 12-8
 - topology 12-8, 12-14
 - Topology Change Notification 12-11
- Spanning Tree Bridge
 - defaults 12-4, 13-3
- Spanning Tree bridge parameters
 - 802.1D standard protocol 12-24
 - 802.1s multiple spanning tree protocol 11-1, 12-24
 - 802.1w rapid reconfiguration protocol 12-24
 - automatic VLAN containment 12-29
 - forward delay time 12-27
 - hello time 12-25
 - maximum age time 12-26
 - priority 12-24
- Spanning Tree Modes 11-11
 - 1x1 mode 11-11
 - flat mode 11-11
- Spanning Tree Port
 - defaults 12-5
- Spanning Tree port parameters 12-30
 - connection type 12-39
 - link aggregate ports 12-33, 12-35, 12-37, 12-38, 12-40
 - mode 12-38
 - path cost 12-35
 - priority 12-34
- specification
 - IPv6 29-2
- Specifications
 - QoS 39-2
- specifications
 - 802.1AB 22-2
 - 802.1Q 24-2
 - dynamic link aggregation 26-2, 27-3
 - Ethernet 1-2
 - Ethernet OAM 17-2, 18-2
 - GVRP 5-2
 - interswitch protocols 23-2
 - IP 28-2
 - Port Mapping 8-2
 - port mirroring 43-4
 - port monitoring 43-6, 43-8
 - RDP 31-2
 - RIP 30-2
 - RMON 43-11
 - Spanning Tree 12-3, 13-2
 - static link aggregation 25-2
 - switch health 43-13
 - switch logging 44-2
 - UDLD 20-2
 - VLAN rules 9-2
- SSL
 - for LDAP authentication servers 36-46
 - policy servers 38-6
- static agg agg num command 25-3, 25-9
- static link aggregation 25-1
 - adding ports 25-9
 - application examples 25-3, 25-11
 - configuration steps 25-7
 - creating 25-8
 - defaults 25-2
 - deleting 25-8
 - deleting ports 25-9
 - disabling 25-10
 - enabling 25-10
 - group names 25-10
 - groups 25-5, 26-6
 - overview 25-5, 26-6
 - specifications 25-2
 - verify information about 25-12
- static linkagg admin state command 25-10
- static linkagg name command 25-10
- static linkagg size command 25-3, 25-8
- static MAC addresses 2-5
- static route
 - IP 28-10, 29-12
 - metric 28-11, 29-12
 - subnet mask 28-10
- static VLAN port assignment 7-4
- subnet mask 28-10
- switch health
 - application examples 43-14
 - defaults 43-14
 - monitoring 43-41
 - specifications 43-13
- switch health statistics
 - resetting 43-48
 - viewing 43-46
- switch logging
 - application examples 44-4
 - application ID 44-6
 - defaults 44-3
 - output 44-9
 - severity level 44-8
 - specifications 44-2

status 44-10
 swlog appid level command 44-6
 swlog clear command 44-11
 swlog command 44-4, 44-6
 swlog output command 39-21
 swlog output command 44-9
 swlog output flash file-size command 44-10

T

TCN BPDU
see Topology Change Notification BPDU
 TCP
 statistics 28-38
 time-to-live
see TTL
 Topology Change Notification BPDU 12-11
 ToS
 trusted ports 39-32
 traceroute command 28-38
 tracking
 VRRP 34-9
 traffic prioritization 39-68
 Transparent Switching 5-8
 trap port link command 1-7
 traps
 port link messages 1-7
 Trust 16-6
 trusted ports
see also ports
 used with QoS policies 39-33
 TTL value 28-19

U

UDLD
 application examples 20-3
 defaults 20-2
 disabling on port 20-6
 disabling on switch 20-6
 enabling on port 20-6
 overview 20-4
 show 20-8
 specifications 20-2
 uddl command 20-3
 uddl port command 20-3
 UDP 28-38
 statistics 28-38
 User Datagram Protocol
see UDP
 users
 functional privileges 36-14, 36-41

V

Vendor Specific Attributes
see VSAs
 Virtual Router Redundancy Protocol
see VRRP
 virtual routers 34-7

vlan 802.1q command 4-7, 4-9, 7-4, 24-5
 vlan 802.1q frame type command 24-6
 VLAN advertisements
 application examples 5-4
 vlan command 5-5, 28-4, 30-3
 vlan dhcp generic command 9-10
 vlan dhcp mac command 9-9
 vlan dhcp mac range command 9-9
 vlan dhcp port command 9-10
 vlan ip command 9-11
 vlan mac command 9-10
 vlan mac range command 9-11
 vlan mobile-tag command 4-8, 7-5
 vlan port 802.1x command 37-9
 vlan port authenticate command 7-16
 vlan port command 9-13
 and 802.1X ports 35-74, 37-4
 vlan port default command 4-6, 4-7, 7-4, 28-4, 30-3
 vlan port default vlan command 7-16
 vlan port default vlan restore command 7-16
 vlan port mobile command 4-7, 7-5, 7-10, 7-11
 vlan protocol command 9-12
 vlan router ip command 28-5
 VLAN rules 9-1, 9-8
 application examples 9-3, 9-14
 defaults 9-2
 DHCP 9-5, 9-9, 9-10
 MAC address 9-5, 9-10
 MAC range 9-11
 network address 9-5, 9-11
 port 9-6, 9-13
 precedence 9-6
 protocol 9-5, 9-12
 specifications 9-2
 types 9-3
 VLAN Stacking
 display list of all or range of configured SVLANs 13-25
 displaying the configuration 10-37
 vlan stp command 4-9
 vlan svlan command 12-21
 VLANs 4-1, 4-5, 13-4
 802.1Q 24-3
 administrative status 4-6
 application examples 4-3, 4-10, 7-3
 default VLAN 7-1, 7-12
 defaults 4-3
 description 4-6
 IP multinetting 28-6
 IP router ports 28-7
 MAC address aging time 2-9
 mobile tag classification 4-8
 operational status 4-5
 port assignment 4-6, 7-1
 rule classification 4-8
 secondary VLAN 7-13
 Spanning Tree status 4-9
 tagging 24-3
 VLAN ID 4-5
 VRRP 34-1

- ACLs 34-10, 34-20
- application example 34-5, 34-28, 34-32
- ARP request 34-8
- backup router 34-7
- defaults 34-3
- MAC address 34-8
- master router 34-7
- tracking 34-9
- virtual routers 34-7
- vrrp command 34-10, 34-20
 - defaults 34-3
- vrrp delay command 34-15
- vrrp ip command 34-10, 34-20
- vrrp track command 34-27
- vrrp track-association command 34-27
- vrrp trap command 34-15, 34-25
- VRRP3 34-20
 - Advertisement Interval 34-22
 - application examples 34-33
 - Preemption 34-23
 - Traps 34-25
 - Virtual Router 34-20
 - Virtual Router Priority 34-23
- VSAs
 - for LDAP servers 36-41
 - for RADIUS authentication 36-10
 - RADIUS accounting servers 36-15
 - setting up for RADIUS servers 35-71, 36-13

W

- warnings 44-8